

DVD ДАРОМ Ubuntu 14.10, openSUSE

LINUX FORMAT

Главное в мире Linux

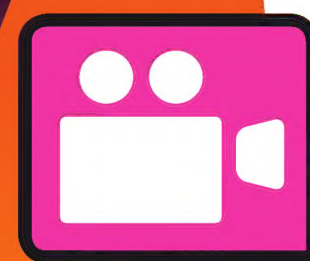
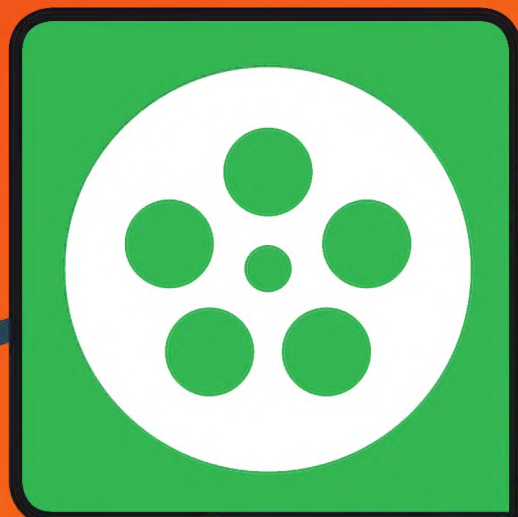
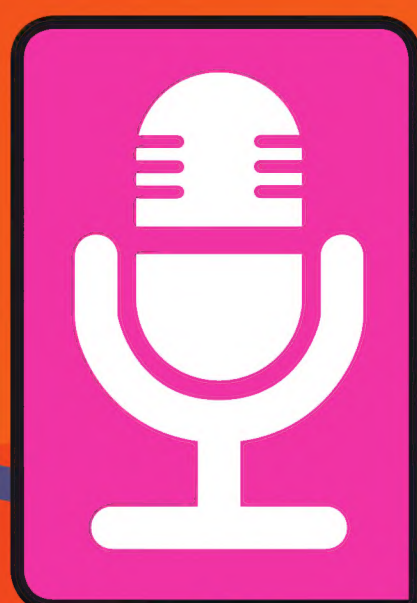
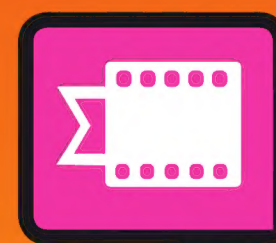
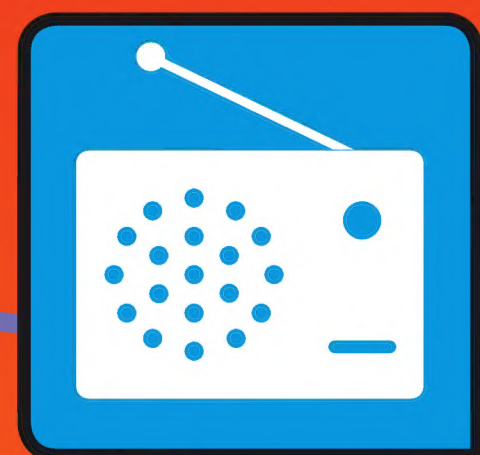
Февраль 2015 № 2 (193)

СТРОИМ МЕДИА-СЕРВЕР LINUX

Превратите свой ПК в хаб — вещайте поток фильмов и музыки на все устройства в доме

Ускоряем
Python

» Как с Cython
ОПТИМИЗИРОВАТЬ КОД



ЕСТЬ
**ЭЛЕКТРОННАЯ
ВЕРСИЯ**
ДЛЯ ВАШЕГО
ПЛАНШЕТА!

ПЛЮС
39 страниц
проектов

» Надежная очистка дисков,
оптимизация брандмауэра
и многое другое!



Ядра

Изучим Minix 3

» Испробуйте ядро,
которое вдохновило Линуса

Файловые системы

Освоим ZFS и btrfs

» Установите передовую
ФС на своем массиве RAID

Minetest

Войдем в моды

» Начините киллера
Minecraft по-своему

ПОДПИСНЫЕ ИНДЕКСЫ В КАТАЛОГАХ
Агентство «Роспечать» — 36343.
«Почта России» — 11932, «Пресса России» — 90959

Linux center
www.linuxcenter.ru



iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ

Что мы делаем

» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.



Кто мы

Если бы сбылась мечта Стива Балмера и открытый код стерли с лица Земли, каковы были бы последствия?



Джонни Бидвелл

Ну, мы все остались бы без работы. Но я не думаю, что открытый код умер бы навсегда. Рано или поздно кому-то захотелось бы делиться — правда, тогда под вопросом было бы и существование Интернета. Как контейнеры по доставке, сетевые протоколы работают лучше при открытых стандартах.



Нейл Ботвик

Это не будет проблемой. Просто вытащите из GitHub предыдущую версию *Планеты Земля* и ответьте ей. Вот и все. Здесь смотреть нечего. Двигайтесь вперед.



Шон Конвэй

Если бы свободное ПО исчезло с лица Земли, мне пришлось бы заново придумывать, чем заниматься, сидя у себя в подвале. Подобно колодцу в пустыне для страждущего путника, открытое ПО — это ассортимент прохладительных напитков для путешественника. Без него пришлось бы пить из крана.



Эндрю Моллетт

По-простому, и нам, и миру пришел бы конец. На открытом коде функционирует Лондонское Сити, финансовый центр мира. Даже если убрать из уравнения финансы, как бы мы учились? Как развивались? OSS — это общность знаний и эволюция, а не только выполнение привычных занятий.



Маянк Шарма

Я могу представить массу последствий (нет Raspberry Pi, нет Linux, нет *Apache*, нет Всемирной паутины), но наихудшим лично для меня было бы отсутствие этого прекрасного журнала. А я, видимо, посещал бы факультет института по эксплуатации пишущих машинок.



Валентин Синецын

Поначалу все бы, конечно, приуныли. Но потом какой-нибудь шустрый финский студент взялся бы писать эмулятор терминала... и все вернулось бы на круги своя.



Время «больших данных»



Как и в мире высокой моды, в публикациях об ИТ периодически возникают те или иные «тренды», причем каждый раз утверждается, что именно эта технология в очередной раз «спасет мир». Сегодня так говорят о big data — обработке гигантских объемов данных, не обязательно структурированных в виде таблиц реляционной СУБД. Последнее — главное отличие от столь же «модных» полтора десятилетия назад технологий data mining.

Технические возможности для этого есть. Емкость носителей растет по экспоненте. По некоторым оценкам, на дисках, изготовленных в прошлом году, можно разместить все цифровые данные, наработанные человечеством за первые 60 лет существования ИТ — примерно зеттабайт. Необходимость работы с такими объемами привела к появлению нового поколения файловых систем (btrfs, XFS) и средств распределенного хранения данных (ceph, GlusterFS).

Первое десятилетие нынешнего века ознаменовалось безоговорочной победой «цифровой революции». Объем цифровых данных минимум на порядок превышает суммарный объем всех «аналоговых» носителей (книг, картин, фото- и киноплёнок), и разрыв стремительно увеличивается.

Можно утверждать, что мы научились хранить и передавать любые требуемые объемы данных. А вот быстро получить из них информацию — еще большой вопрос...

Кирилл Степанов

Главный редактор

» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru

Подписка и предыдущие номера: subscribe@linuxformat.ru

Техническая поддержка: answers@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

Сайт: www.linuxformat.ru, группа «ВКонтакте»: vk.com/linuxform

» Адрес редакции: Россия, Санкт-Петербург, пр. Медиков, 5, корп. 7

» Телефон редакции: (812) 309-0686. Дополнительная информация на с. 112

Знаете всё о Linux?

ДА

НЕТ

Станьте автором
в журнале Linux Format!
linuxformat.ru/avtoram.phtml

Зарегистрируйтесь
на сайте
shop.linuxformat.ru



СОМНЕНИЯ

Скачайте бесплатно
архивные PDF-номера
журнала с сайта
linuxformat.ru/archive
и загляните на
wiki.linuxformat.ru



Версия для iPad
и iPhone доступна
в App Store

Выберите вид подписки

PDF-версия
на 6 месяцев

990 ₽

PDF-версия
на 12 месяцев

1800 ₽

Печатная версия
на 6 месяцев

1890 ₽

Печатная версия
на 12 месяцев

3480 ₽

Выберите вид доставки

Оплатите

Читайте Linux Format!

Станьте Linux-гуру

PDF-версия журнала Linux Format подойдет для тех, кто:

- Заботится о соблюдении прав деревьев
- Любит читать с экрана
- Мечтает получать каждый номер в день выхода журнала
- Хочет бесплатно скачивать содержимое DVD-приложения к каждому номеру

Печатная версия Linux Format понравится читателям, которые:

- Любят читать бумажные журналы
- Хотят получить в подарок подписку на PDF-версию Linux Format...
- ...а также диск с архивом журнала 2005–2014
- Порадуются новинкам открытого ПО на DVD-приложении к Linux Format в каждом номере

Способы доставки

- Курьером «ГНУ/Линуксцентра» по Москве и Петербургу
- Курьерской службой СПСР по России
- Почтой по России заказной или простой бандеролью
- Самовывоз из офиса «ГНУ/Линуксцентра» в Санкт-Петербурге
- Через пункты выдачи интернет-магазинов iml.ru в 11 городах России: Санкт-Петербург, Москва, Екатеринбург, Калуга, Нижний Новгород, Орел, Ростов-на-Дону, Тверь, Тюмень, Челябинск, Ярославль

Способы оплаты

- По квитанции в любом отделении Сбербанка
- Яндекс.Деньги, Webmoney
- Пластиковой картой Visa/MasterCard
- Наличными в офисе «ГНУ/Линуксцентра»
- Безналичный (для юридических лиц)



«ГНУ/Линуксцентр»

Санкт-Петербург,
пр. Медиков, 5, корп. 7
(метро «Петроградская»)

(812) 309-0686

www.linuxformat.ru

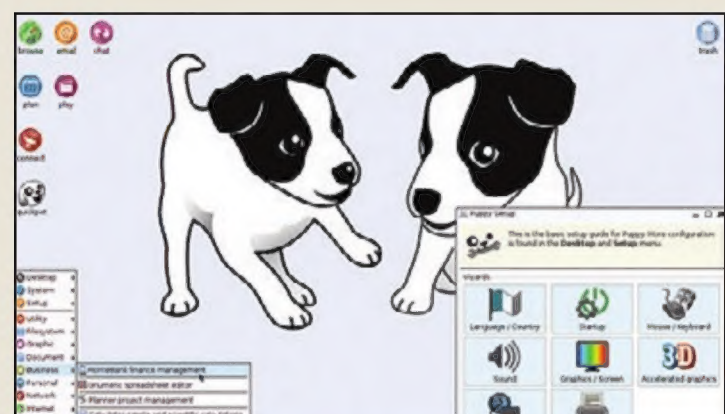
Содержание

«Поборемся с неизъяснимым и посмотрим, не удастся ли его изъяснить.»

Обзоры

Puppy Linux 6.0 14

Быстрый и легкий дистрибутив приведет в восторг новичков и владельцев старых компьютеров.



➤ Резвый и дружелюбный, буквально как щенок.

Cinnamon 2.4 15

Новая версия рабочего стола вознаградит его фанатов, но особых сюрпризов не ждите: разработчики просто перетряхнули разболтавшийся код.

Vaultier 0.7 CE 16

О менеджерах паролей свое отговорили; разберемся, заслуживают ли внимания хранилища паролей.

Hover 17

Придайте вашему ПК антуража «Особого мнения» при помощи этой недорогой платы для управления жестами.

Google Nexus 6 18

Первый телефон на Android 5.0 Lollipop — красивая штука; а удовольствия-то сколько! Особенно удивил непривычно четкий дисплей, на зависть даже iPhone.

Google Nexus 9 19

Первый фаблет от Google на Lollipop, напротив, сравнительно неказист, но зато добротный работяга.

Chromebook 2 от Samsung 20

Хромбук — тонкий и элегантный, и одновременно по доступной цене? Уж не колдовство ли это?

Deronia 21

Ленивый раздолбай — герой этой игры — пытается сбежать с помойки своей жизни... и обретает любовь.

Игра и книга 22

Football Manager 2015 нас не воспламенил, а вот «Приключения в *Minecraft*» очень даже заинтриговали. Они наверняка понравятся детям — приятное с полезным...

Строим медиа-сервер Linux

Превратите свой ПК в хаб потокового вещания фильмов и музыки на все устройства в доме с. 32



Сравнение: Редакторы векторной графики с. 26



Люди говорят



« В первой версии моторы были приклеены прямо к плате »

Команда Bitcraze обсуждает открытые нанокоптеры с. 40

На вашем бесплатном DVD



Ubuntu 14.10
XBMCbuntu 13.2
openSUSE 13.2

» Убойные дистрибутивы по теме номера

ПЛЮС: HotPicks и коды учебников **с. 106**

Побалуйте себя и любимых
подпиской на LXF!



Доступно
в AppStore!



www.linuxformat.ru/subscribe

Ищите в этом номере

Соберем NAS 44

От нуля до 24 ТБ — объясняем, как построить классное сетевое хранилище для обслуживания файлов.

Освоим ZFS и btrfs 50

Выберите для своего RAID одну из двух на шумевших файловых систем нового поколения.

Моды Minetest 54

Покопаемся в открытой игре-песочнице, навеянной *Minecraft*, пользуясь ее API и модами.



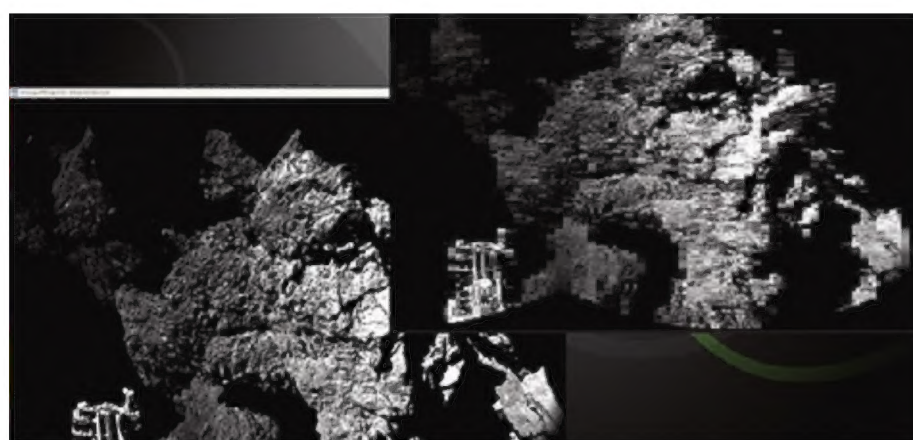
Академия кодига

Cython: Даешь скорость 88

Ускоряем код на Python на примере алгоритма сжатия изображений (всё потому, что мы поленились просмотреть фотоотчеты Миссии Розетта).

R: Как много в этой букве 92

Введение в язык программирования для статистики, с целью применить его для анализа данных о системе. Он такое вытворяет с диаграммами, что у вас глаза на лоб полезут.



Постоянные рубрики

Новости 6

Роса сертифицировалась на HP, Google задумался о Linux, а Canonical — об Интернете вещей, побратались «ТОНК» с Альт Линукс и Oracle с Samsung, вышел российский аналог Azure, WebOS нашла себе нишу, а Торвальдс опять спорит.

Новости Android 24

На Android пришла Adobe, Google Play поборол Apple App Store, а LibreOffice уже в телефоне — как и 3D-печать.

Сравнение 26

Мы бросили вызов пятерым лучшим свободным пакетам для векторной графики, заставив их пройти наши суровые тесты — без сомнений, колебаний и пикселизации. *Inkscape, Karbon, LibreOffice Draw, SK1, Xara Xtreme.*

Интервью LXF 40

Наноквадрокоптер команды **Bitcraze** из Швеции помогает разыскивать пропавших в горных районах Итальянских Альп.

Рубрика сисадмина 58

Доктор раскрывает суперкомпьютерные секреты британских метеорологов и объясняет, как настроить сервер OpenLDAP на хранение данных об учетных записях пользователей.

Ответы 96

ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!

Нейл Ботвик — про сортировку фотографий, надежное удаление данных с флэшек, работу с «чужеродными» дисками, безопасность и шифрование облака, решение проблем с btrfs, копирование MP3 по порядку.

HotPicks 100

Отведайте горяченького! Лучшие в мире новинки свободного ПО: *0 A.D., CherryTree, Konversation, Lunar Linux, Neverball, OpenSSH, Remmina, rTorrent, Snapper, Tomahawk, Wal Commander.*

Диск Linux Format 106

Содержимое двустороннего DVD этого месяца.

Пропустили номер? .. 108

Купите немедленно, чтобы смягчить свою вину за упущение! Никакой комплект не полон без **LXF190**.

Через месяц 112

Под нашим руководством, откройте для себя самые передовые графику и файловую систему и настройте Linux нового поколения.

Учебники

Grub

Меню a la carte 64

Заставьте меню загрузки плясать под вашу дудку, настроив его внешность и пункты: средства позволяют.

Minix 3

Тест микроядерной ОС 66

Испробуйте ядро, послужившее источником вдохновения для Linux. Мы научим, как запустить его в *VirtualBox*.

Жесткие диски

Стереть бесследно 70

Протестируйте 7 методов стирания конфиденциальных данных со старых дисков — от *DBAN* и *Secure Erase* до винтовки Ли-Энфилда!

Ipset

Упростим брандмауэр 74

Сделайте управление брандмауэром менее мучительной задачей, применив *ipset* — расширение *Netfilter*. Читать и редактировать файлы настройки станет значительно проще.

Awk, sed

Обработка текстов 78

Выуживайте критически важные сведения из файлов журналов автоматически, вооружившись нашими ценными советами по созданию файлов Awk.

ПЛИС

Елочка, гори 82

Новый год вернется опять! Подготовимся к его встрече заранее: соберем мерцающую гирлянду светодиодов. Вообще-то на Новом году свет клином не сошелся — есть и другие праздники...

В ЭТОМ НОМЕРЕ: Роса на HP » Google Drive и Linux? » Интернет вещей » Импорт-замещение » Виртуализация » Облака сгущаются » Умные вещи » Мнения Линуса

ОС + СЕРВЕР = ...

Успешная сертификация

ОС РОСА SX «КОБАЛЬТ» и РОСА SX «ХРОМ» протестированы на серверах HP класса Superdome X.

В конце 2014 года компания HP и российский разработчик защищенных ОС «ИТЦ ИТ РОСА» успешно протестировали работоспособность серверных ОС РОСА «КОБАЛЬТ» и «ХРОМ» на сервере HP Integrity Superdome X, с учетом особенностей его архитектуры.

Испытания проходили в Центре высоких технологий офиса HP в Москве. В состав тестового стенда входил сервер HP в следующей конфигурации: 120 физических ядер Intel Xeon E7 (240 логических ядер с технологией Intel Hyper-Threading); 3 ТБ ОЗУ; SAN 16 Гб; сеть 10 GE. В рамках испытаний тестировались доступ к сервисам организации через web-интерфейс, средства управления рабочими местами,

средства документооборота, интеграции в корпоративные сети, виртуализации, подключения к внешним хранилищам SAN, агрегирование сетевого трафика.

«Идеальная платформа для критичных к отказам приложений.»

Сервер HP Integrity Superdome X был представлен 2 декабря 2014 г., это одна из новейших и значимых разработок компании. Наряду с высокой масштабируемостью и производительностью, данный сервер обладает уникальными для мира

x86 надежностью и отказоустойчивостью, заимствованными от модельного ряда HP Integrity Superdome 2; это идеальная платформа для ресурсоемких и критичных к отказам приложений корпоративного уровня, работающих в окружениях Linux индустриального стандарта. В максимальной конфигурации сервер поддерживает до 16 процессоров Intel Xeon E7 (до 240 вычислительных ядер) и 12 ТБ ОЗУ (модулями DIMM по 32 Гб).

Функциональные и нагрузочные тесты успешно выдержали все подсистемы: процессоры, память, сетевой ввод-вывод. Подсистема виртуализации распознала все предоставленные ей ресурсы и работала в штатном режиме.



» Рубрику готовил
**АНДРЕЙ
ГОНДАРЕНКОВ**

ДОЖДАЛИСЬ?

О необходимости которого...

В Сеть просочились скриншоты Google Drive для Linux.

Вот уже три года, с момента запуска файлового хранилища Google Drive, пользователи GNU/Linux ждут появления официального клиента для этой операционной системы. За минувшее время корпорация Google успела, например, разработать и представить на рынок 3 версии своей мобильной платформы Android. Что же мешало IT-гиганту выпустить для Linux нативный продукт, который есть даже у таких небольших игроков на этой площадке, как ownCloud, Seafiler, Dropbox или Cору?

Однако, господа присяжные заседатели, очень похоже, что лёд наконец-таки

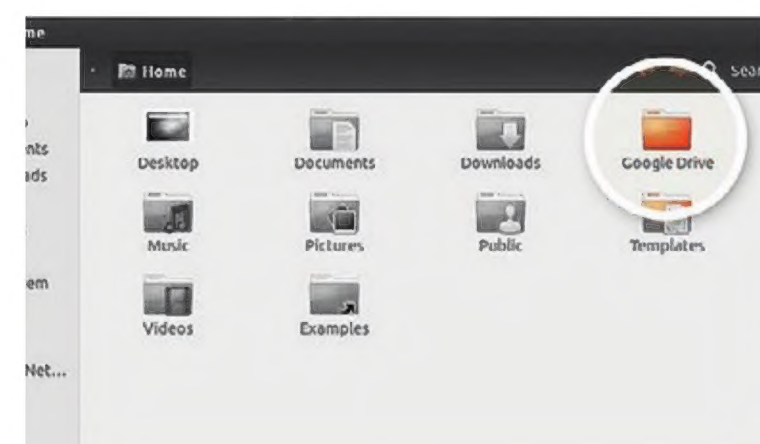
тронулся: в Сети появились скриншоты, на которых виден клиент Google Drive, запущенный в Ubuntu.

Обладатели машин Mac, скачавшие клиент Google Drive (версия 1.18.7821.2489 для MacOS), установившие его и выполнившие поиск по запросу 'lin.png', найдут два файла — 'setup_taskbar_lin.png' и 'setup_drive_folder_lin.png' — на которых отображён запущенный в Linux клиент Google Drive.

Возникает естественный вопрос о достоверности полученных изображений. Не Photoshop ли это? Однако, учитывая, что источником информации является

» То самое изображение из Mac-файла setup_drive_folder_lin, подтверждающее наличие Linux-версии клиента Google Drive.

сам Google (Google Drive.app/Contents/Resources/lib/python2.7/resources/images/), можно предположить, что тестирование клиента Google Drive для Linux всё же проводится, и рано или поздно мы получим официальный релиз.



ОС ВОКРУГ НАС

Ubuntu Core для Internet of Things

Canonical выпустил новый продукт для умных устройств.

Марк Шаттлворт представил Snappy Ubuntu Core — новый продукт Canonical, предназначенный для работы с «умными» потребительскими устройствами — такими словами: «„Умные“ устройства приносят новое содержание в наши дома, работу и игры. Изобретатели и новаторы создают невероятные машины — роботов, выполняющих уборку или перемещающих вещи по дому; дронов, замещающих нас в играх; умные дома, эффективно использующие энергию или предоставляющие более информативные системы безопасности. Подавляющая часть этих технологий использует Linux — но это очень фрагментированный и небезопасный вид Linux. Каждое „умное“ устройство имеет собственную „прошивку [firmware]“, объединяющую в единое целое ОС, драйверы и специфичное для устройств программное обеспечение. При этом такие „прошивки“ практически никогда не обновляются. Давайте исправим ситуацию!»

Сочетание формирующей единый образ базовой ОС без пакетных менеджеров системы Snappy и Ubuntu Core идеально подходит для распределенных подключенных устройств, которые нуждаются в обновлениях системы безопасности для ОС и приложений, и должны быть также абсолютно надежными и самовосстанавливаемыми. Snappy избавляет распределенные устройства от необходимости удовлетворять пакетные зависимости.

Партнерская экосистема Snappy Ubuntu Core объединила 22 участника: смарт-хабы Ninja Sphere от Ninjablocks, ОС ROS от OSRF, фреймворк Openhab, дрон Eerle-Copter от Eerle Robotics; производителей плат Odroid, Beaglebone, Banana Pro, Udoo, PCDuino, Parallella и вендора чипов Allwinner; решения для Интернета вещей (IoT) Kaa, DeviceHive, 2lemetry, IoTSys, Resin.io, OpenSensors.io; открытая ОС для IoT Riot-OS; системы сетевой коммуникации для IoT Nwave и Fairwaves; сеть Weave для Docker; судно-робот Trasibot.



➤ Марк Шаттлворт выступает на презентации нового продукта.

Новый продукт поддерживает архитектуры ARMv7 и x86-64. Для Ubuntu Core необходим процессор с тактовой частотой 600 МГц и 128 МБ ОЗУ, из которых система использует 40 МБ. Для сброса к заводским параметрам и отката системы устройства должны иметь 4 Гб flash. Для настройки системного окружения и управления приложениями предлагается специальный web-интерфейс WebDM.

НАШЛИ ДРУГ ДРУГА

Союз аппаратуры и программ

ГК «ТОНК» и ALT Linux стали стратегическими и технологическими партнерами.

Генеральные директора Группы компаний «ТОНК» и компании ALT Linux, Михаил Ушаков и Алексей Смирнов, подписали соглашение о глобальном технологическом партнерстве. С января 2015 г. крупнейший российский производитель тонких клиентов и ведущий в стране разработчик свободного ПО начинают совместную разработку терминалов, работающих под управлением операционных систем на основе ядра Linux.

Системы на базе аппаратных решений ТОНК и ПО ALT Linux смогут полноценно заменить проприетарные средства виртуализации от зарубежных производителей. Так, многофункциональный дистрибутив для серверов и рабочих станций Альт Линукс Кентавр 7.0 поддерживает виртуализацию в среде KVM на платформе x86 по протоколу SPICE.

«Наши разработки позволяют российским заказчикам полностью решить

проблему, связанную с заменой продукции компаний, присоединившихся к санкциям иностранных государств против России, — подчеркнул Михаил Ушаков. — Разработанные и произведенные в России, прошедшие сертификацию ФСТЭК

«Смогут полноценно заменить ПО от зарубежных производителей.»

системы смогут обеспечить и необходимую функциональность, и требуемый уровень информационной безопасности. Нелишний довод в пользу наших совместных с ALT Linux решений — и их полное соответствие новым требованиям продукции в рамках закупок для госнужд».

В ноябре и декабре 2014 года специалисты компании ALT Linux провели мас-

штабное тестирование всех продуктов Группы компаний «ТОНК». Оно показало полную совместимость тонких клиентов ТОНК и безопасных компьютеров LXBOX серий 2 и 3 с программными продуктами семейства Альт Линукс СПТ 7.0.

В настоящее время в ФСТЭК завершается процедура сертификации новых версий продуктов компании ALT Linux, которыми будут комплектоваться различные системы ТОНК: Альт Линукс СПТ 7.0 Рабочая станция (обеспечивает комфортную защищенную работу с информацией ограниченного доступа с возможностью просмотра и копирования данных меньшей степени конфиденциальности или из открытых источников), система для бездисковой рабочей станции Альт Линукс СПТ 7.0 Тонкий клиент и Альт Линукс СПТ 7.0 Сервер, сочетающий все возможности классического сервера с современными технологиями управления.

ВИРТУАЛЬНЫЕ РЕСУРСЫ

Облакоотека на новой платформе

Запущена новая платформа AzuRus с линейкой тарифных планов WAZ.

В рамках стратегии импортозамещения отечественная компания Облакоотека запустила новую платформу AzuRus, представленную линейкой новых тарифных планов WAZ. Услуга представляет собой виртуальную IT-инфраструктуру с web-интерфейсом управления AzurePack, аналогичным интерфейсу управления Azure.

Особенности новой услуги:

- » Привычный и удобный web-интерфейс
- » Размещение ресурсов в российском ЦОД
- » Виртуальные сети с NAT
- » Построение Site-to-Site VPN
- » Подключение к консоли виртуальных машин
- » Добавление со-администраторов к управлению

Платформа AzuRus представляет собой виртуальную инфраструктуру,

развернутую на площадке Облакоотеки на базе платформы Hyper-V, с управлением всеми виртуальными ресурсами из web-интерфейса, полностью аналогичного интерфейсу Azure, и является полнофункциональной платформой для раз-

«Состав ресурсов определяется тарифным планом.»

мещения приложений. В рамках услуги создается полноценное частное облако с predetermined количеством виртуальных ресурсов. Состав ресурсов определяется соответствующим тарифным планом. В случае необходимости в режиме реального времени можно арендовать дополнительные виртуальные ресурсы —

расширить существующее облако либо перейти на старший, более затратный тарифный план.

В услугу входят:

- » облако виртуальных машин (набор ограниченных ресурсов) для создания IT-инфраструктуры, размещенное на площадке Облакоотеки;
- » виртуальные сети, возможность их создания и управления ими, включая построение NAT и Site-to-Site VPN;
- » подключение к виртуальной машине через консоль;
- » web-интерфейс AzurePack для управления виртуальными машинами в Облакоотеке;
- » все необходимые возможности самостоятельного управления ресурсами;
- » возможность предоставления дополнительного доступа IT-специалистам для управления инфраструктурой.

СЕРВИСЫ

Облака: Тандем гигантов

Oracle и Samsung объединят усилия для создания мобильного облачного сервиса.

Как сообщает The Korea Times, на прошедшей в конце января в Сеуле встрече сопresident Oracle Марк Херд [Mark Hurd] и глава мобильного подразделения Samsung Electronics Шин Йонг-Кюн [Shin Jong-Kyun] обсудили детали партнерства, в итоге которого решения для облачных баз данных Oracle обогатятся возможностями Samsung для корпоративных клиентов. Подобно недавней сделке IBM—Apple, этот союз, нацеленный на удовлетворение потребностей корпоративных мобильных пользователей, будет выгоден обоим участникам. Представители Oracle и Samsung от комментариев отказались.

Во II квартале прошлого года доходы Oracle от ПО и облачных сервисов выросли на 5%, до \$7,3 млрд, а доходы от облачных сервисов SaaS, PaaS и IaaS выросли на 45%, до \$516 млн. Samsung уже имеет опыт партнерства с Microsoft и SAP.

«Учитывая стремительное проникновение мобильных устройств на территорию корпоративного сектора, ориентирование

технологий на потребителя является одним из ключевых факторов мотивации, — отметил Рэй Вонг [Ray Wang], основатель и главный аналитик Constellation Research. — При этом многие организации пока только пытаются понять, как управлять всеми этими устройствами».

Партнерство с Samsung не только позволит Oracle расширить область использования своего программного обеспечения, но и предоставит возможность выйти на рынок потребительских това-

«Перспектива партнерства Oracle и Samsung интригует.»

ров. На свои устройства Samsung предлагает цены, более чем доступные для развивающихся рынков. В то же время, маржа Samsung от корпоративного сектора значительно превышает получаемую от устройств для потребительского рынка.

» Подобно недавней сделке IBM — Apple, это партнерство, нацеленное на удовлетворение потребностей корпоративных мобильных пользователей, будет выгодно обоим участникам.



Однако пока неясно, сможет ли Oracle убедить своих клиентов воспользоваться основанными на облачных вычислениях мобильными сервисами.

«Традиционное ПО от Oracle глубоко интегрировано во многие бизнесы, но его облачные усилия и стратегии менее очевидны, и компания выглядит опаздывающей на фоне более агрессивных игроков, — считает Чарльз Кинг [Charles King] аналитик Pund-IT. — В целом, хотя перспектива партнерства Oracle и Samsung, безусловно, интригует, сейчас существует намного больше вопросов, чем ответов на них».

Добро пожаловать в робототехнику!

ScratchDuino

Электронный комплекс на основе свободного аппаратного обеспечения для школ и вузов

ScratchDuino.Лаборатория

Плата расширения с датчиками ввода-вывода — как установленными на плате, так и подключаемыми. Предназначена для программирования взаимодействия компьютера с внешними устройствами.

Полная интеграция («из коробки») со средой программирования Scratch, предназначенной для детей.

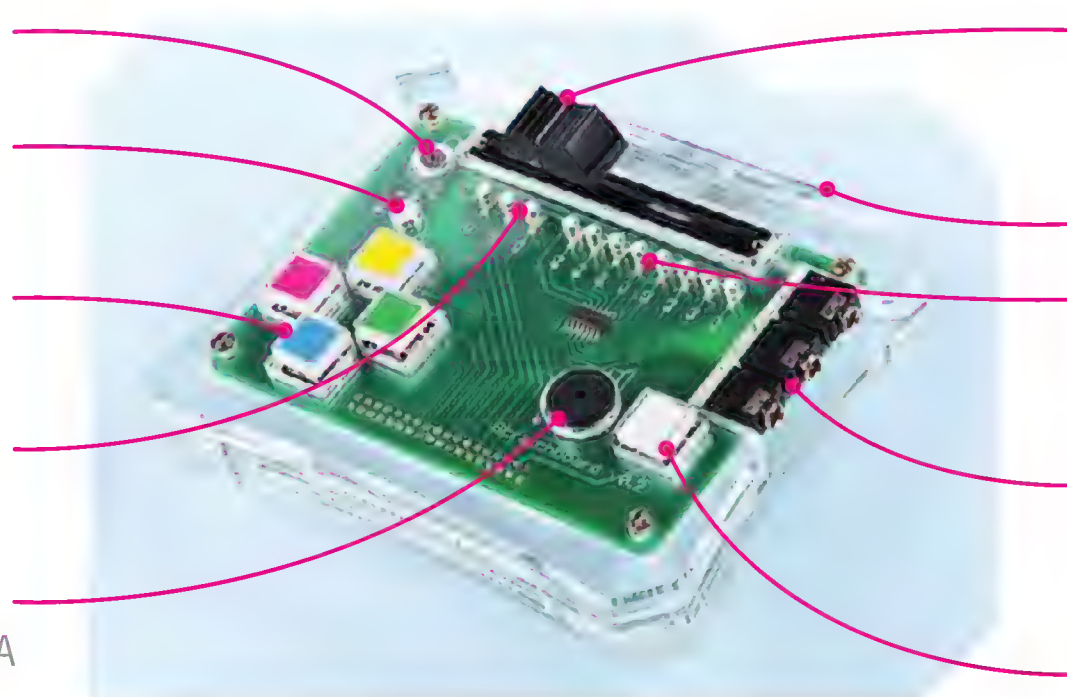
ДАТЧИК ЗВУКА*

ДАТЧИК ОСВЕЩЕННОСТИ*

КРЕСТОВИНА КНОПОК

ЦВЕТНЫЕ СВЕТОДИОДЫ

ДИНАМИК
ДЛЯ ВЫВОДА ЗВУКА



ПЕРЕМЕННЫЙ РЕЗИСТОР (ПОЛЗУНОК)*

КАРТРИДЖ ARDUINO

СВЕТОДИОДЫ, УПРАВЛЯЕМЫЕ ЧЕРЕЗ РАЗРЯДНУЮ МАТРИЦУ

РАЗЪЕМЫ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ

КНОПКА*

* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ RISCBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **9500 руб.**

ГАРАНТИЯ 3 ГОДА,
ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ
И МЕТОДИЧЕСКУЮ
ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа

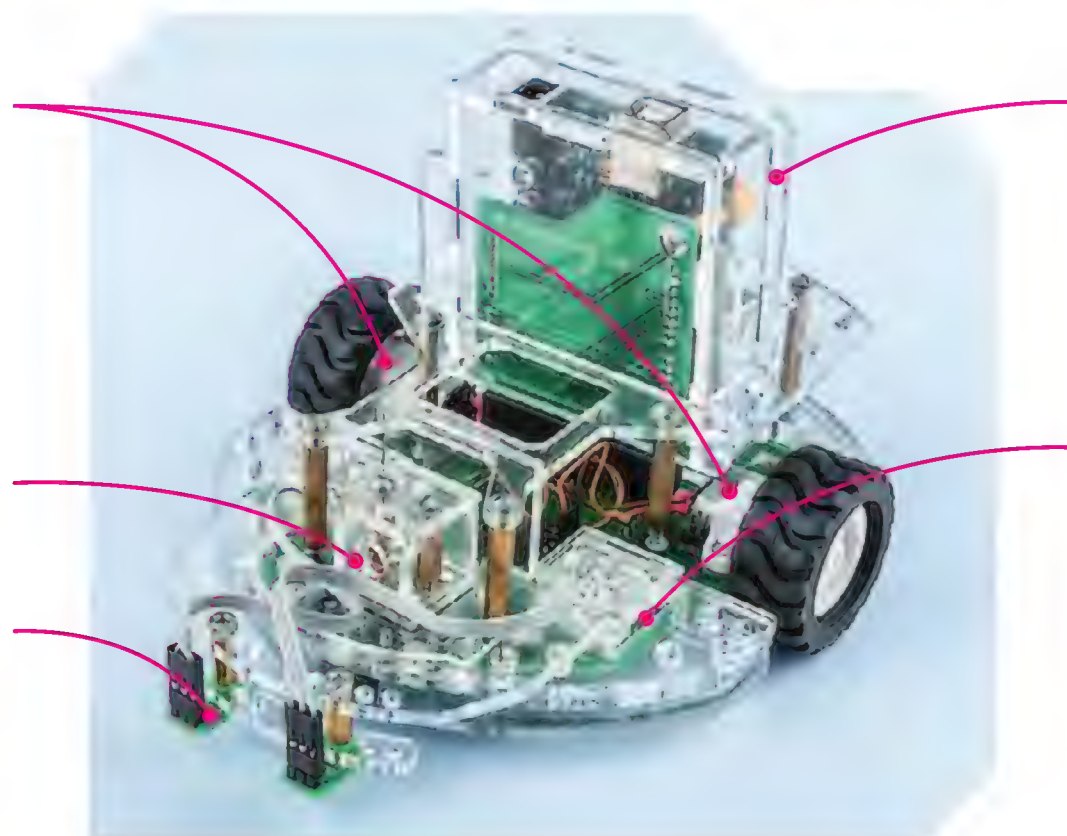
обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.

МОТОР-РЕДУКТОРЫ

ФОТОДАТЧИК

ДАТЧИК ЛИНИИ



КАРТРИДЖ ARDUINO

МАГНИТНЫЙ ЦИФРО-АНАЛОГОВЫЙ РАЗЪЕМ
ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ БЕЗ МЕХАНИЧЕСКИХ СОЕДИНЕНИЙ И ПАЙКИ*

* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **19 500 руб.**

ГАРАНТИЯ 3 ГОДА,
ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ
И МЕТОДИЧЕСКУЮ
ПОДДЕРЖКУ

Назначение ScratchDuino

- Изучение взаимодействия компьютера с внешней средой
- Изучение изменения параметров внешней среды
- Изучение процессов передачи информации и принципов ее построения
- Изучение внешних устройств управления
- Моделирование устройств

Среда разработки

Lazarus (язык Pascal) ■ Scratch ■ Arduino IDE

ScratchDuino адаптирован для учебных заведений. Поставляется с комплектами учебно-методических материалов.

Продукт разработан при финансовой поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере

WWW.SCRATCHDUINO.RU

(812) 309-0686 (Санкт-Петербург) ■ (499) 271-4954 (Москва)

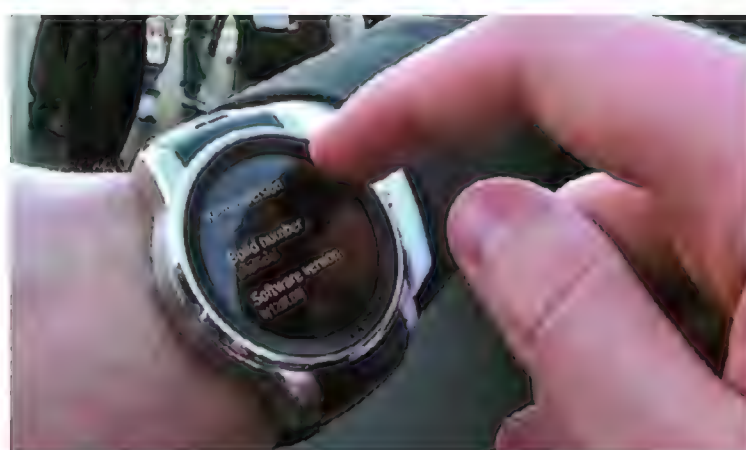
«УМНЫЕ ВЕЩИ»

WebOS бросает вызов

Почти забытая ОС попала в «умные часы» LG-Audi.

Для своей совместной разработки пока безымянных «умных» часов LG и Audi неожиданно выбрали WebOS, хотя ранее планировалось применить для этой цели Android Wear (версия для носимых устройств). WebOS была в 2009 г. представлена Palm как проприетарная; затем её приобрела HP, а в 2013 г. — LG, адаптировавшая её для своих Smart TV. Сейчас код WebOS открыт, и она пытается поколебать доминирование Google в сегменте потребительских smart-устройств. Новый гаджет содержит полный комплект полагающихся телефону приложений (сообщения, галерея, музыка, электронная почта). На выставке потребительской электроники Consumer Electronics Show показали только прототип, но он уже теперь выглядит серьёзным конкурентом: например, укомплектован динамиком, отсутствующим у большинства альтернатив.

Поскольку одним из создателей новых часов является Audi, неудивительно



» Управляемое WebOS водонепроницаемое устройство из нержавеющей стали и сапфирового стекла является серьёзную угрозу конкурентам.

наличие у них приложения для разблокировки автомобиля через NFC. Также присутствует Q Voice, голосовой помощник от LG. А вот наличие приложения 'Find My Phone' невольно вызывает вопрос о продуманности выбора ПО для новинки; хотя, возможно, это просто рудимент WebOS, от которого разработчики забыли избавиться.

Часы выполнены из нержавеющей стали и сапфирового стекла и сделаны водонепроницаемыми. Цена нового устройства пока не объявлена, но применённые материалы и брендинг Audi намекают на ценник из разряда «премиум».

ЛИНУС СКАЗАЛ...

Создатель Linux — о себе

А также о диверсификации в мире Linux и о выявленных проблемах безопасности.

На состоявшейся в середине января в Окленде (Новая Зеландия) конференции linux.conf.au Линус Торвалдс заявил, что сообщения о выявленных проблемах безопасности абсолютно необходимы и должны появляться «в разумные сроки». Линус оспорил утверждение, что раскрытие подобных проблем только помогает т. н. «чёрным» хакерам: наоборот, подобная политика стимулирует разработчиков к скорейшему закрытию уязвимостей. Торвалдс отметил, что список рассылки ядра Linux сообщает о проблемах безопасности в течение 5 рабочих дней. «В других проектах это может быть месяц или несколько месяцев. Что гораздо лучше многих лет молчания, как раньше».

Ряд высказываний создателя Linux на сессии Q&A конференции вызвал негативную реакцию у членов Сообщества. На замечание разработчика Nebula (и бывшего коллеги) Мэттью Гарретта [Matthew

Garrett] о зачастую колком и агрессивном тоне Линуса в официальных списках рассылки тот ответил: «Я человек неприятный, и люди меня не интересуют. Меня интересуют технологии и ядро — вот что важно для меня». Проблему диверсификации усилий разработчиков Linux Торвалдс назвал «мелочью, реально не важной».

Позднее новостной и аналитический портал Ars Technica поместил письмо Линуса с более подробным изложением его взглядов, в т. ч. и на «благопристойность» в секторе Open Source. Популярную в США концепцию «Ты должен быть приятным человеком» Линус назвал «идеологической»: «Некоторые разработчики привыкли к конфронтационному стилю и предпочитают его». Торвалдс добавил, что «плохие идеи необходимо агрессивно отклонять, но и хорошие идеи нуждаются в энергичной защите», хотя честно признался: «Возможно, я просто люблю поспорить». LXF

Новости короткой строкой

» Открытый гипервизор Xen 4.5 поддерживает набор инструкций SMAP новых процессоров Intel и расширения DBE для новых процессоров AMD. Источник: blog.xenproject.org

» Прокси-сервер Squid 3.5 реализовал поддержку именованных сервисов, eCAP 1.0 и протокола PROXY, портировал из Squid-2.7 опцию collapsed_forwarding, обновил утилиту squidclient. Источник: marc.info

» Вышел Btier 2.0.0 — модуль ядра Linux для формирования многоуровневых блочных устройств из разнотипных накопителей небольшого объёма. Источник: www.lessfs.com

» В OrientDB 2.0 (открытой документо- и графо-ориентированной СУБД): новая панель Security для управления пользователями и ролями; модуль ETL для импорта данных из JDBC, CVS и JSON; улучшены автовыделение динамической памяти и дискового кэша. Источник: www.orienttechnologies.com

» Бывший глава Opera Software Йон фон Течнер [Jon von Tetzchner] представил прототип кроссплатформенного (Windows, GNU/Linux, MacOS X) web-браузера Vivaldi на движке Chromium 40. Источник: thenextweb.com

» Выявленная в библиотеке GNU C Library (glibc) критическая уязвимость (кодовое имя GHOST) позволяет реализовать удалённое выполнение кода (RCE) и получить контроль над сервером под управлением GNU/Linux. Источник: openwall.com

» В LibreOffice 4.4 обновили диалог выбора цвета и шаблоны; реорганизованы панели инструментов, контекстные меню и диалоги, введены новые шрифты Carlito и Caladea, улучшен импорт форматов OOXML, Adobe Pagemaker, Microsoft Visio и Abiword. Источник: blog.documentfoundation.org

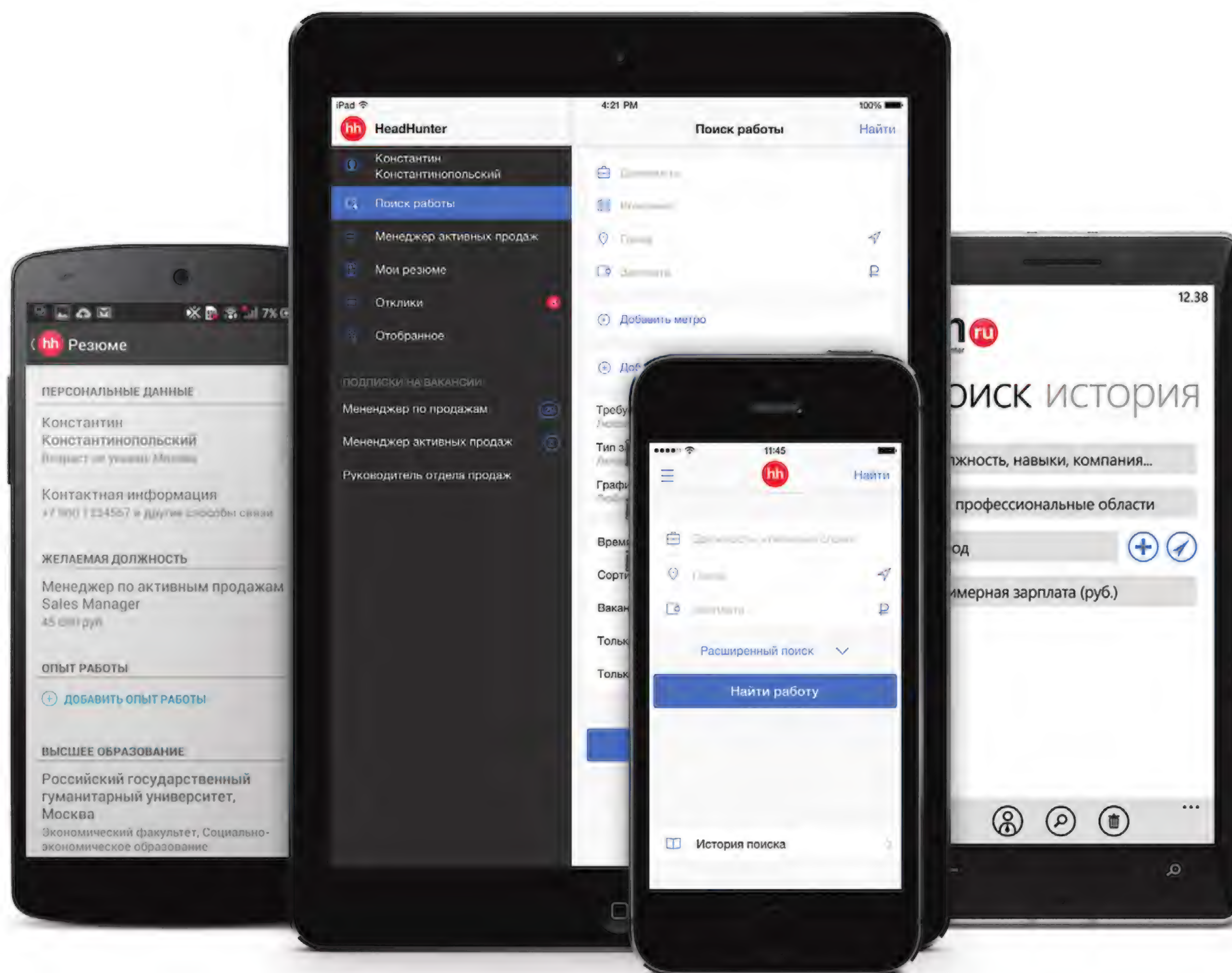
» В Inkscape 0.91 — новая система отрисовки; улучшены инструменты работы с градиентами и текстом, потребление памяти и совместимость с форматами Corel DRAW и Visio. Источник: inkscape.org

» В Chrome, IE 11, Safari 8 и beta-версии Firefox хостинг YouTube теперь по умолчанию отображает видеоконтент проигрывателем HTML5 с тэгом <video>. Источник: youtube-eng.blogspot.ru

0+



Работа мечты в твоём кармане!

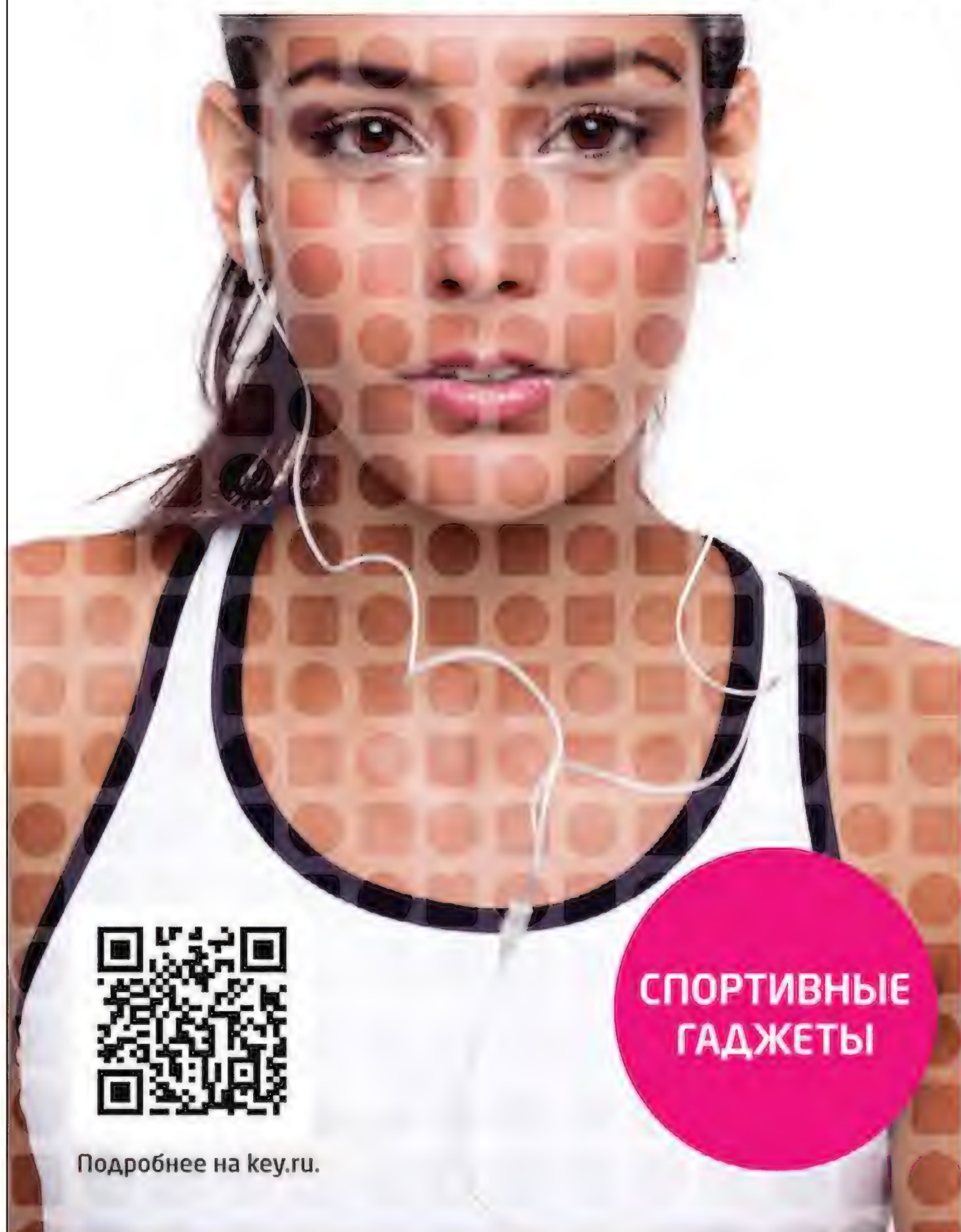


Искать просто, откликаться легко.

<http://hh.ru/mobile>



ТВОИ ПОМОЩНИКИ В ТРЕНИРОВКАХ



Экшн - камера



Экшн-камера Sony HDR-AS30

Smart - часы



Smart-часы Cookoo Watch

Smart - здоровье



Браслет Jawbone UP



Шаромер Misfit Shine



Товар сертифицирован. Количество товара ограничено.

СПОРТИВНЫЕ
ГАДЖЕТЫ



Подробнее на key.ru.

Одноплатный компьютер Cubieboard3

Cubietruck



ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

- Процессор: AllWinnerTech SOC A20, ARM Cortex-A7 Dual-Core
- Видео: ARM® Mali 400 MP2
- Выходы: HDMI, VGA 1080p
- Оперативная память: 2 ГБ, DDR3, 480 МГц
- Встроенная память: 8 ГБ NAND
- Внешние накопители: интерфейс SATA 2.0 для 2,5" HDD; возможно подключение 3,5" HDD с использованием внешнего питания 12 В
- Сеть: 10M/100M/1G Ethernet, Wi-Fi и Bluetooth
- Питание: DC 5 В, 2,5 А с подключенным HDD; возможно подключение

литиевого элемента питания

- Периферия: 2 × USB host, 1 × USB OTG, 1 × Toslink (SPDIF), 1 × IR, 4 × LEDs, 3,5" аудиовыход, 3 кнопки
- Поддерживаемые ОС: Android, Linux-based, BSD
- 54 контакта расширения I2S, I2C, SPI, CVBS, LRADC × 2, UART, PS2, PWM × 2, TS/CSI, IRDA, LINEIN & FMIN & MICIN, TVIN × 4
- Размер платы: 11 × 8 см

КОМПЛЕКТАЦИЯ

- Плата Cubietruck
- Прозрачный акриловый корпус
- Кабель питания USB
- Кабель MiniUSB
- Кабель SATA с питанием
- Радиатор для процессора

www.linuxcenter.ru/shop/ops_hard/cubieboard/cubietruck/

«ГНУ/Линуксцентр»: +7 812 309-06-86 (Санкт-Петербург), +7 499 271-49-54 (Москва)



LINUX FORMAT Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов



АЛЕКСЕЙ ФЕДОРЧУК
Тэг <сарказм>
по умолчанию,
смайлики по вкусу.

Cinnamon: начало триумфального шествия?

В прошлой колонке речь шла о релизе среды Cinnamon 2.4, вошедшей в состав одноимённой редакции Mint 17.1. Но значение его не только в этом.

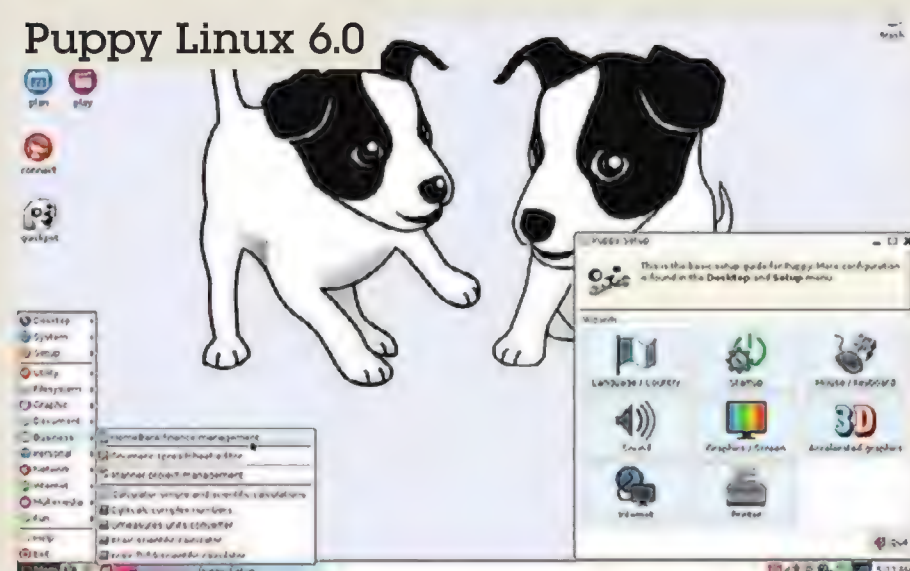
Не секрет, что до недавнего времени Cinnamon поддерживался майнтейнерами более иных дистрибутивов не лучшим образом. Фактически, единственной полностью работоспособной сборкой этой среды за пределами прародительской Mint была та, что в PPA-репозитории Ubuntu поддерживал Gwendal Le Bihan. Да и тот прошлой весной отказался от развития стабильной её ветки, ограничившись «ежегощными» тестовыми. В прочих же популярных дистрибутивах поддержка Cinnamon была либо чисто формальной, либо не очень аккуратной, либо просто не совсем полной. И стало казаться, что Cinnamon уготована судьба десктопа одного дистрибутива... сами знаете кого и где. Но с релизом среды 2.4 всё кардинально изменилось.

Во-первых, сам Gwendal продолжал регулярно выпускать тестовые сборки — как задел на будущее. Во-вторых, на их базе образовалось аж два стабильных PPA-репозитория — Lester Carballo Pérez и Tsvetko. В-третьих, и главных: на 100% рабочие сборки Cinnamon 2.4 появились для таких столпов, как openSUSE, Fedora (и Russian Fedora), Archlinux и его Manjaro, и даже для PC-BSD (это только опробованные лично). Не начало ли это триумфального шествия данной среды по пользовательским ПК — давнишняя мечта её поклонников?

alv@posix.ru

Сегодня мы рассматриваем:

- | | | |
|---|---|--|
| Puppy Linux 6.0 14 | Hover 17 | Deponia 21 |
| Редчайший случай в наших Обзорах: дистрибутив, заработавший идеальную «десятку», без малейших изъянов. С удовольствием придрались бы, да не к чему... | Для Raspberry Pi создана плата расширения, управляемая жестами и касаниями. Легким движением руки... | Герой этого квеста пытается прогнаться... нет, не через тернии к звездам, а из помойки в рай. Не очень-то он и достоин рая, но по ходу многое осознать. |
| Cinnamon 2.4 15 | Google Nexus 6 18 | Football Manager 2015 22 |
| Усовершенствования данной версии набирающего популярность рабочего стола в основном упрятаны «под капотом»; внешние же изменения в глаза не бросаются. | В новом фableте от Google Android научился делать в фоне больше действий, чем когда-либо. И прочие характеристики тоже на высоте. | Если вы — счастливый обладатель версии этой игры от 2014 г., расслабьтесь: обновляться смысла нет. Остальным... ну, можно и поиграть. |
| Vaultier 0.7 CE 16 | Google Nexus 9 19 | Книга о Minecraft 22 |
| Это хранилище паролей, с RSA-шифрованием по ключу, установить будет непросто: потребуется тонкая настройка системы. Не боязно ли доверить ему свои секреты? | Google предпринял вылазку в «верхний ценовой сегмент» планшетов, где царит iPad. Дебют многообещающий. | Имя Мартина О'Хэнлона в LXF упоминается достаточно часто, обычно в связи с Minecraft. И вот Мартин, в соавторстве с Дэвидом Уэйлом, выпустил книгу о своих проектах. |
| | Chromebook 2 20 | |
| | У тонкого и элегантного хромбука от Samsung непривычно яркий дисплей и батарея-долгожитель. | |



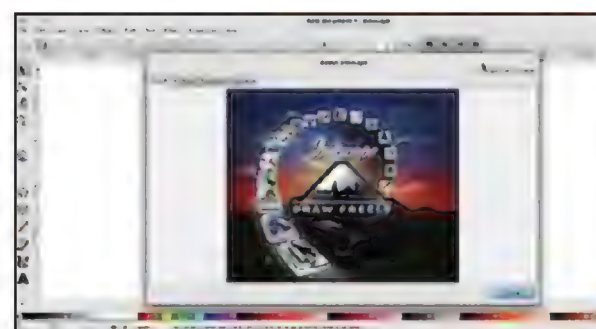
➤ Вид резвящихся щеночков вызывает непроизвольную улыбку. Тонкий расчет?



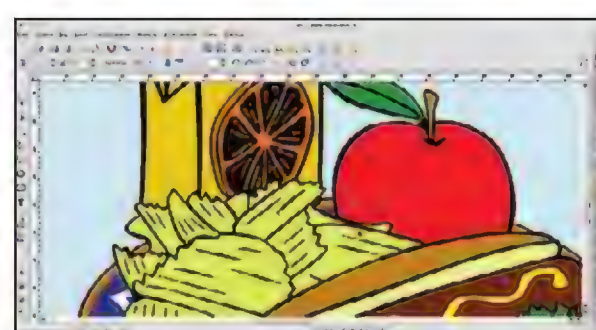
➤ Сколько уж там оттенков серого у корпусов — кто его знает, а на экране цвет очень живой.

Сравнение: Редакторы векторной графики с. 26

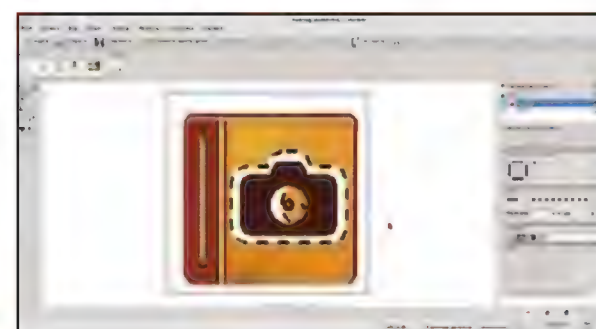
Inkscape



sk1



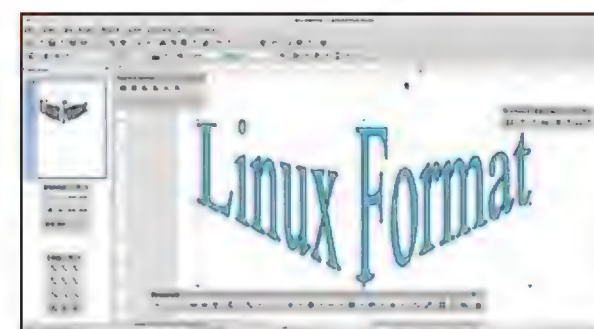
Karbon



Xara Xtreme



LibreOffice Draw



Прелесть векторной графики — в масштабируемости: как ни варьируй размер векторного изображения, качество его не ухудшится. Забудьте про пикселизацию! (Если сами не опуститесь до пикселей.)

Puppy Linux 6.0

Неужели все обзоры Puppy обязаны упоминать пёсиков? Попросив лапу у свежей версии этого друга человека, **Шашанк Шарма** дерзнул пойти против системы.

Вкратце

» Легковесный дистрибутив для старых компьютеров и систем со слабыми ресурсами. Содержит несколько собственных графических инструментов для настройки системы, что делает его идеальным для новичков. См. также: Tiny Core Linux, Lubuntu.

Может быть, дело в названии, а может, в команде разработчиков, но Puppy [англ. «щенок»] Linux, похоже, овладел искусством всем нравиться. Со времён своего дебюта 11 лет назад этот легковесный дистрибутив выучился новым трюкам, но по-прежнему жизнерадостен, быстр, дружелюбен и надёжен.

Последний Puppy Linux, названный Tahrpup, поставляется как live-образ с возможностью установки. Проект предлагает два образа по 199 МБ — с поддержкой PAE и без неё. Первый вариант нужен для машин с более чем 4 ГБ памяти. Дистрибутив использует оконный менеджер JWM и даёт быстрый доступ ко всем приложениям, которые аккуратно разделены по категориям, таким как Личное [Personal], Утилиты [Utilities], Интернет [Internet], Мультимедиа [Multimedia]. Большинство легковесных дистрибутивов также и минималистичны, но Tahrpup порой предоставляет приложения с расширенной функциональностью. Это наиболее заметно в меню Мультимедиа и Интернет, где вы найдёте несколько музыкальных плееров и приложений для обмена файлами.

Клиент электронной почты в дистрибутиве — Sylpheed, а браузер по умолчанию — Pale Moon, основанный на Firefox. Flash по умолчанию не установлен, но браузер умеет работать с потоковым видео. Для работы с офисными документами по умолчанию включены AbiWord и Gnumeric. Также дистрибутив содержит утилиту Pup Advert Blocker для защиты от рекламы.

Дистрибутив предлагает быструю установку популярных приложений. С помощью утилиты GetFlash вы можете установить официальный плагин Adobe Flash. Можно быстро загрузить и установить LibreOffice с помощью утилиты из меню Документы [Document].

Несмотря на малый объём, дистрибутив полон приложений, причём некоторые из них нетрадиционны. Здесь есть HomeBank, что позволит вам управлять финансами, GWhere для каталогизации дисков, менеджер паролей Figaro's Password Manager и графические инструменты для управления сетевыми дисками Samba и настройки брандмауэров.

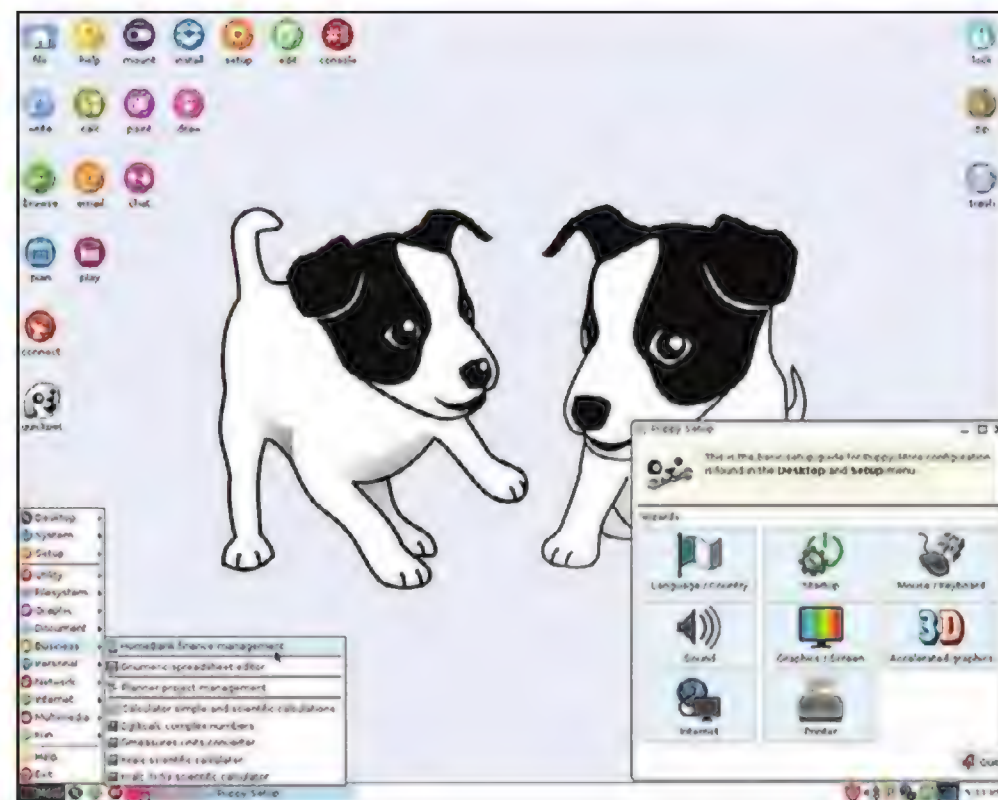
Двоичная совместимость с Ubuntu 14.04 означает, что если вы не обнаружите в этом внушительном списке любимое приложение, можно воспользоваться средством управления пакетами для доступа к репозиториям Ubuntu и легко установить добавочные пакеты. А для быстрой установки ряда наиболее популярных программ пригодится утилита QuickPet.

Настройка Tahrpup

Изначальная конфигурация Puppy не подключена к Интернету. При первом запуске Puppy вам предложат настроить сеть, и в этом процессе вам помогут несколько собственных инструментов настройки.

В Puppy входят уникальные инструменты и графические мастера настройки, которые облегчают управление сетью, графическими дисплеями, принтерами и так далее. Инструменты дружелюбны к новичкам, а мастера предоставляют подробные инструкции, помогая выполнить все действия. Независимо от того, запустили ли вы Puppy с USB или с CD, когда вы решите закончить live-сессию, вам предложат сохранить изменения. Если вы согласитесь, Puppy сохранит в указанном месте все ваши настройки, предпочтения и изменения в системе.

В дистрибутив входит средство управления дисками GParted, которое сыграло свою роль, если вы решите установить Puppy на диск. Установка с помощью



» Один из самых быстрых дистрибутивов среди виденных нами. А также самый аккуратный, самый милый и... ну хватит вилять хвостом!

мастера здесь одна из простейших и на каждом шаге предлагает достаточно информации для принятия решения, что делать дальше. В качестве загрузчика Tahrpup использует Grub4dos и позволяет при необходимости вносить изменения в конфигурационные файлы во время установки. На рабочем столе представлены несколько иконок для часто используемых приложений. Но вместо названий программ иконки названы по предоставляемым функциям. Так, иконка «написать [write]» откроет AbiWord, а «воспроизвести [play]» — VLC.

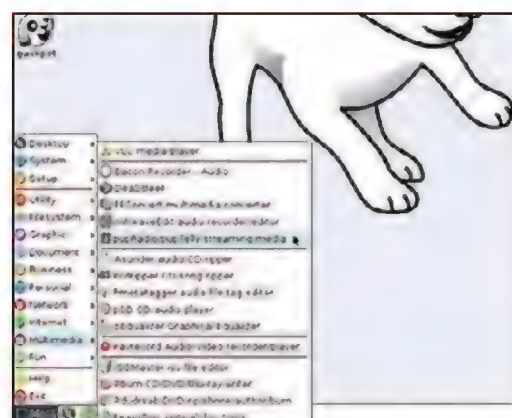
Как мы ни старались, нам так и не удалось найти у этого щенка блох. Puppy Linux — один из простейших в использовании дистрибутивов, независимо от уровня ваших навыков. Мы решительно рекомендуем его новичкам, ищущим веселенький дистрибутивчик. **LXF**

Свойства навскидку



Собственные инструменты

Вы можете легко изменить различные аспекты системы через меню настроек.



Коллекция приложений

Дистрибутив подходит широкому ряду пользователей и содержит все популярные приложения.

LINUX FORMAT Вердикт

Puppy Linux 6.0

Разработчик: Команда Puppy Linux
Сайт: www.puppylinux.org
Лицензия: GPL/LGPL

Функциональность	10/10
Производительность	10/10
Удобство в работе	10/10
Документация	10/10

» Редко встретишь столь идеальный дистрибутивчик. Процесс настройки упрощают собственные инструменты.

Рейтинг **10/10**

Cinnamon 2.4

Это ответвление от Gnome 3 Shell уже созрело и стало самостоятельной средой. **Маянк Шарма** оценил последнее обновление.

Вкратце

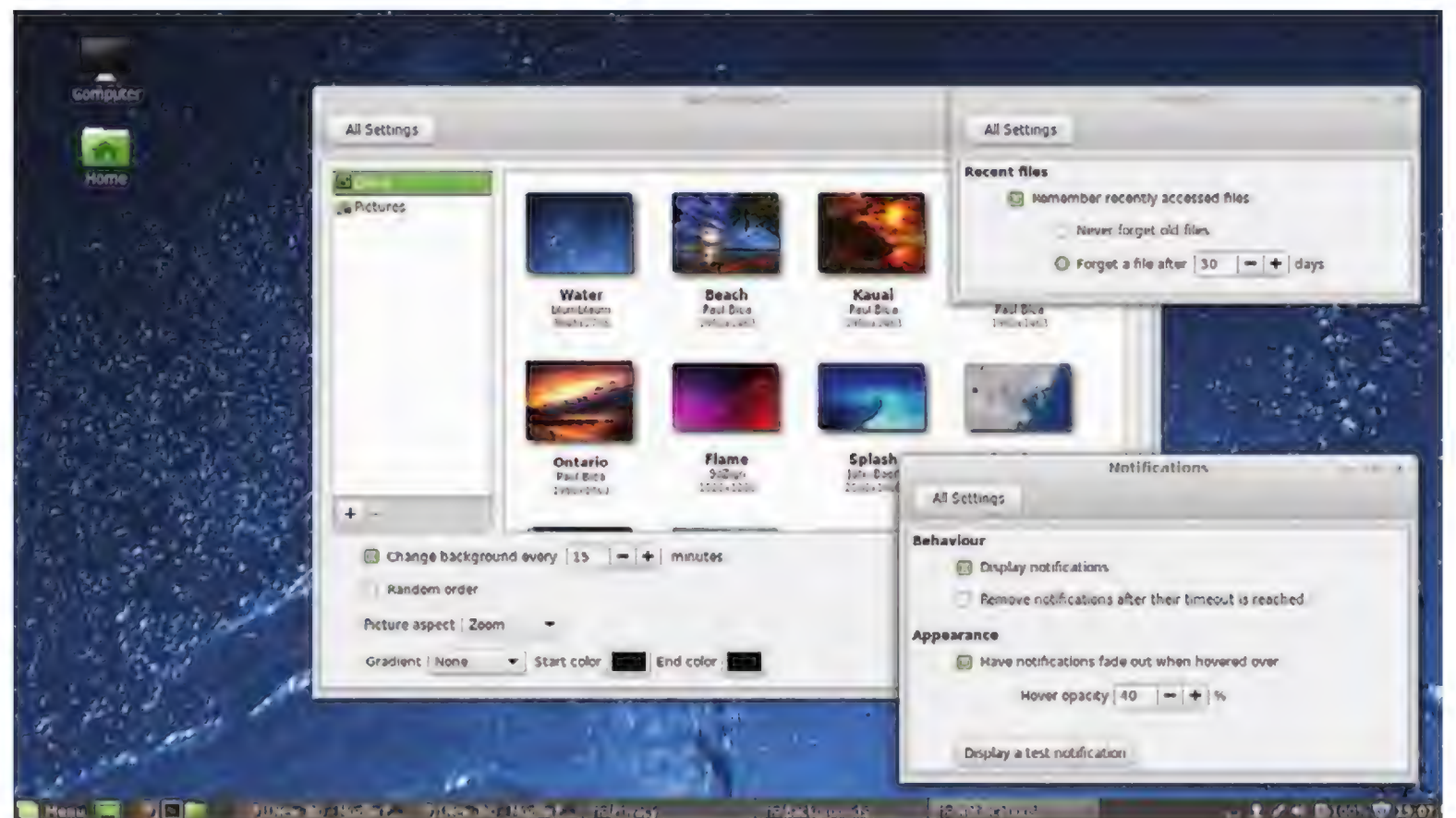
» Ответивший-ся от Gnome Shell рабочий стол на базе GTK3. См. также: Gnome 3 и Pantheon из elementary OS.

Рабочий стол Cinnamon достиг зрелости. В дополнение к Linux Mint, который и ведёт его разработку, он стал популярен среди пользователей других дистрибутивов, мечтающих об окружении, которое основано на последних библиотеках, но придерживается традиционной метафоры рабочего стола. За три года существования Cinnamon вырос из ветки в отдельную рабочую среду, обзаводясь новыми функциями с каждым релизом.

Все базовые компоненты уже размещены по местам, удобство доведено до блеска, и теперь разработчики сфокусировались на расширении списка возможностей. Свежий выпуск включает много функций, которые Cinnamon одолжил у других популярных рабочих сред, а порой и у проприетарных операционных систем.

Например, представлено переработанное окно «Фоновые рисунки», способное делать слайд-шоу из фоновых рисунков, как в Windows 8. Эта особенность идёт вместе с новым десклетом для управления фоновым слайд-шоу. Кроме этого, появилось расширение neto-emblems, позволяющее добавлять значки к любому контенту, отображаемому файловым менеджером *Nemo*. В самом *Nemo* улучшена боковая панель, обновлён дизайн панели инструментов и добавлен пункт контекстного меню для запуска терминала в текущей директории.

Теперь, как в Windows, добавлено сочетание Super+E для открытия домашней



» Вы можете попробовать Cinnamon 2.4 в Linux Mint 17.1 Rebecca.

директории. Также появилась поддержка однокнопочных координатных площадок-тачпадов, как у MacBook, и вы можете настроить действия для многопальцевых касаний. У Gnome Shell среда одолжила эффект плавного увеличения при запуске и окно сетевых настроек.

Релиз с генеральной уборкой

Cinnamon 2.4 усовершенствовал интерфейс: изменён вид некоторых модулей Системных настроек, чтобы ими было проще управлять. Так, в параметрах тем теперь новый механизм выбора темы. Настройки панелей отныне позволяют разместить несколько панелей, и выбрать размер иконок и текста можно отдельно для каждой из них. Появилось и новое окно «Конфиденциальность», где можно попросить среду не сохранять список недавно открывавшихся файлов или удалять его через определённое время. Кроме того, можно управлять видом и поведением уведомлений в соответствующем разделе.

Разработчики выполнили ещё некоторые теньевые изменения и провели статический анализ кода и его компонентов, что помогло исправить ряд ошибок и около 30 утечек памяти. Были реализованы мелкие улучшения и изменения, такие как удаление счётчика в окне выхода. А ещё теперь вы можете настроить композитинг в режиме полного экрана, не перезагружая рабочую среду, и в менеджере композитинга улучшена производительность игр.

На первый взгляд, в Cinnamon 2.4 не заметно особых отличий от предыдущей версии, и если вы не испытывали проблем с производительностью в прошлых релизах, вы на сей раз не ощутите резкого ускорения. Однако разработчики хорошо потрудились над невидимыми нам проблемами в коде, а в видимой части сделали изменения хоть и малозаметные, но весьма желанные.

Cinnamon 2.4 выпущен скорее не для привлечения новых пользователей, а для удовлетворения уже существующих, и это немалочисленное сообщество. По сложившейся традиции Linux Mint, Cinnamon выпускается примерно за месяц до выпуска того дистрибутива, в который он войдёт (для Cinnamon 2.4 таковым будет Linux Mint 17.1); загрузить его с сайта проекта можно уже сейчас. **LXF**



Свойства навскидку



Обновлённые настройки

В Системных настройках была усовершенствована пара модулей, а также добавлены новые.



Пользователям комфортнее

Помимо обычного исправления ошибок, некоторые компоненты стали удобнее.

LINUX FORMAT Вердикт

Cinnamon 2.4

Разработчик: Команда Linux Mint
Сайт: <http://cinnamon.linuxmint.com>
Лицензия: GPL v2

Функциональность	8/10
Производительность	8/10
Удобство в работе	8/10
Документация	7/10

» Рядовое обновление популярной рабочей среды, продолжающее оттачивать удобство использования.

Рейтинг 8/10

Vaultier 0.7 CE

Хранить секреты бывает трудно, сообразил **Шашанк Шарма**, тестируя наше общее хранилище паролей *Vaultier Community Edition*.

Вкратце

» Хранилище паролей с возможностью совместной работы и сохранения файлов, разработанное для очень специфического типа пользователей и требующее тонкой настройки системы. См. также: *Teampass*, *MyPasswords*.

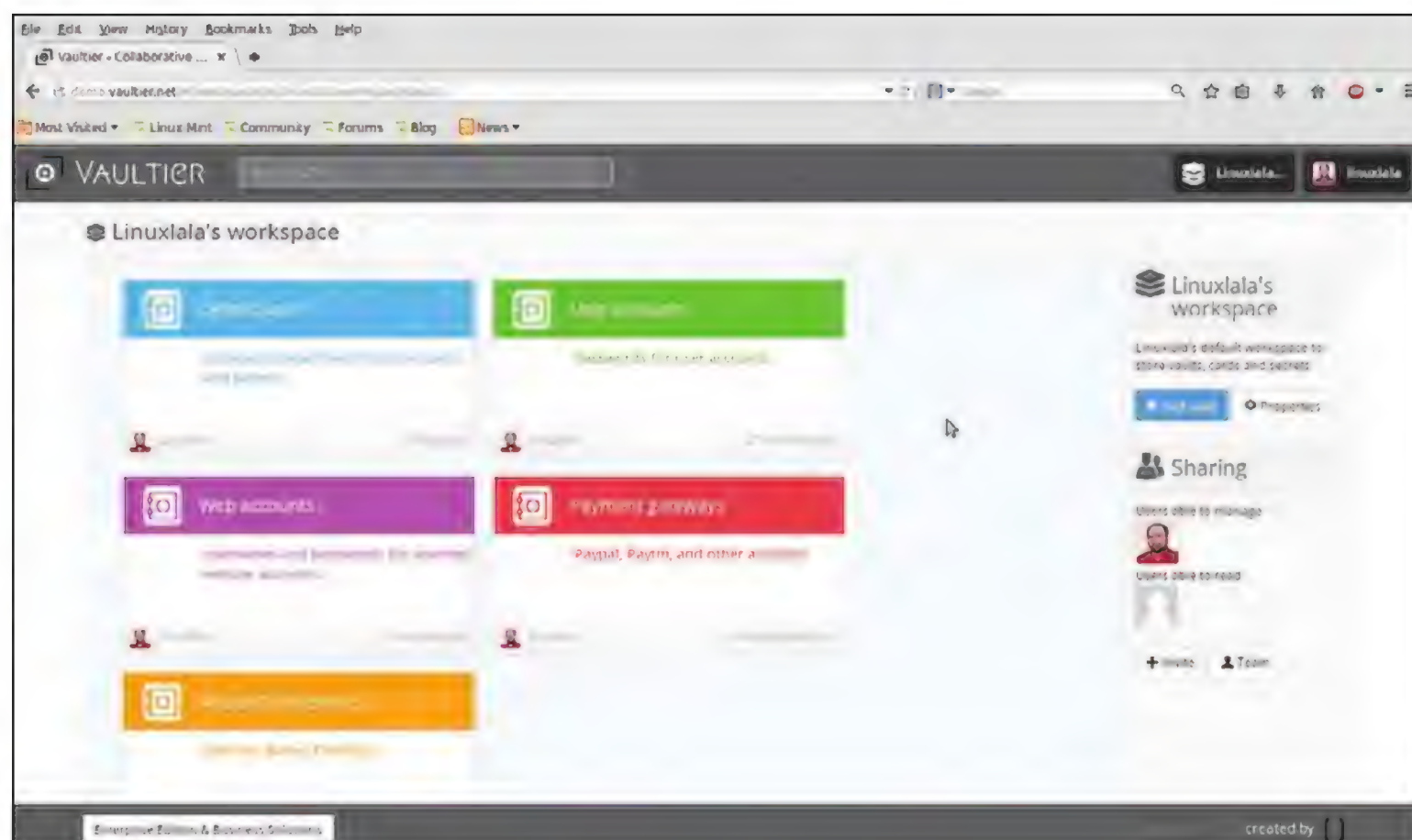
После многолетнего застоя, когда мы видели, как многие инструменты перестали активно развиваться, хранилища паролей вновь возвращаются в моду. Новое поколение, включая *Vaultier*, хвастается такими возможностями, как хранение файлов и сотрудничество пользователей.

Проект *Vaultier* предлагает две версии: Community Edition [для сообщества] с лицензией BSD и версию Enterprise [корпоративная] с тремя планами подписки разной стоимости.

Для Community Edition проект предоставляет три режима установки. Его можно установить в контейнер Docker или с помощью *pip*, установщика Python, а пользователи Ubuntu могут использовать консольный скрипт (предназначенный для Ubuntu 14.04 Server).

Независимо от выбранного режима установки, проекту необходима длительная настройка. *Vaultier* требует базу данных *PostgreSQL* и web-сервер *Nginx* и использует *uWsgi* для работы приложения. Документация проведёт вас через настройку их всех, но это очень запутанный процесс, и придётся самим править файлы настроек. Стандартная документация даёт только самый минимум инструкций, а если вы захотите настроить *Vaultier* для работы с *MySQL* или *Apache*, то вы предоставлены сами себе. У проекта нет даже почтовой рассылки, форума или FAQ для решения проблем, связанных с установкой.

Vaultier заявляет, что очень безопасен, так как использует RSA-шифрование. Когда вы в первый раз создадите учётную



» *Vaultier Community Edition* позволяет быстро и систематизированно хранить пароли, заметки и файлы, но удовольствие от использования сводит на нет сложная установка.

запись для работы с *Vaultier*, он сгенерирует для вас секретный ключ, который надо хранить на локальном диске. В *Vaultier* нет паролей к учётным записям, так что нужно будет указать ему на ключ на компьютере, с которого входите.

Если настроить — хорош

Интерфейс довольно прост, и инструкции помогут вам в деле сохранения секретов и паролей. Информация, будь то пароли, заметки или файлы, которые вы хотите защитить, хранится в виде карточек. Эти карточки находятся в сейфах [vaults], а сами сейфы существуют внутри пространств [workspaces]. Вы можете создать несколько пространств, несколько сейфов в каждом пространстве и несколько карточек в каждом сейфе.

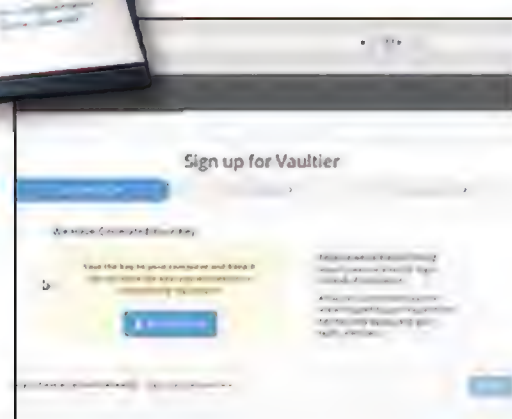
На каждую карточку в сейфе можно занести несколько секретов (паролей, файлов и т.д.). Для сохранения пароля есть форма, где вы указываете сайт, имя пользователя и пароля. Размер файлов *Vaultier* ограничивает 25 КБ.

Если вы продолжите создавать пространства/сейфы/карточки для всех своих секретов, быстрый их поиск может стать проблемой. К счастью, панель поиска в верхней части интерфейса предоставляет быстрый доступ к хранящейся информации. Поиск осуществляется по всем пространствам, сейфам и карточкам и быстро показывает все релевантные вхождения. С механизмом пространств/сейфов/карточек поиск становится неоценимым преимуществом.

Вы можете работать совместно с другими пользователями над определёнными сейфами или над целым пространством. В последнем случае другие пользователи получают доступ ко всем сейфам пространства. *Vaultier* позволяет задать разрешения для каждого члена команды. Можно разрешить им только просмотр данных или выдать им права создавать сейфы и карточки и приглашать новых пользователей.

Но несмотря на эти возможности и интуитивный интерфейс со множеством полезных подсказок, способствующих эффективной работе, мы не можем рекомендовать *Vaultier Community Edition* домашним пользователям из-за сложного процесса его установки и строгих требований. **LXF**

Свойства навскидку



Шифрование ключом

Vaultier не применяет пароли, используя вместо них систему входа по ключу безопасности.



Простой поиск

Поиск осуществляется по именам и описаниям во всех пространствах, сейфах и карточках.

LINUX FORMAT Вердикт

Vaultier 0.7 CE

Разработчик: RightClick

Сайт: www.vaultier.org

Лицензия: BSD

Функциональность	9/10
Производительность	8/10
Удобство в работе	9/10
Документация	5/10

» Если *Vaultier* не упростит процесс установки, есть риск, что его пользователи сбегут к альтернативам вроде *Teampass*.

Рейтинг **7/10**

Hover

Добавьте в свой проект управление жестами и касаниями. Рассел Барнс оценивает близость ощущений от этого набора за £32 к «Особому мнению».

Вкратце

» Комплект для управления жестами и касаниями для ваших проектов на Raspberry Pi, Arduino, pcDuino или Spark Core.

Первым подобное пижонство показал Том Круз в «Особом мнении», и Роберт Дауни-младший до сих пор пытается превзойти произведённый эффект: похоже, в киноиндустрии считают, что нас хлебом не корми, но дай пообщаться с компьютером, жестикулируя перед ним, как маньяк.

Эти манящие прогнозы о будущем человеко-компьютерного взаимодействия пока не совпали с реальностью, но вы можете всего за £32 немного глотнуть будущего с помощью Hover — компактной 6-сантиметровой квадратной платы для разработки, совместимой со многими одноплатными компьютерами и микроконтроллерами, такими как Raspberry Pi и Arduino.

Задумка проста: вы проводите рукой вверх, вниз, влево или вправо в нескольких сантиметрах от платы (сайт говорит о 5 дюймах [12,7 см], но для нас комфортным пределом было около 19 см), и плата регистрирует ваши действия. Мы говорим не о медленных сосредоточенных движениях — подойдёт простое движение запястья в нужном направлении.

Также, если вы любите потыкать свою технику, плата может регистрировать прикосновения. У неё есть пять чувствительных областей: центральная и окружающие её верхняя, нижняя, левая и правая.

Хотя вам и придётся запрограммировать циклы if или for самостоятельно (разработчик Hover Labs обещает обновления в библиотеке, которые это упростят), плата полностью готова регистрировать двойные касания, а также многопальцевые

события. Короче, хитроумные комбинации жестов и касаний скоро будут так и фонтанировать кодом.

Создатели определённо заслуживают аплодисментов за то, что плата совместима с широким набором платформ. Большинство подобных плат работают лишь с Arduino и имеют рудиментарную библиотеку на Python для энтузиастов Raspberry Pi, а у Hover есть подробные инструкции по установке и примеры кода не для одной, а аж четырёх платформ: Raspberry Pi, Arduino, pcDuino и менее известной Spark Core.

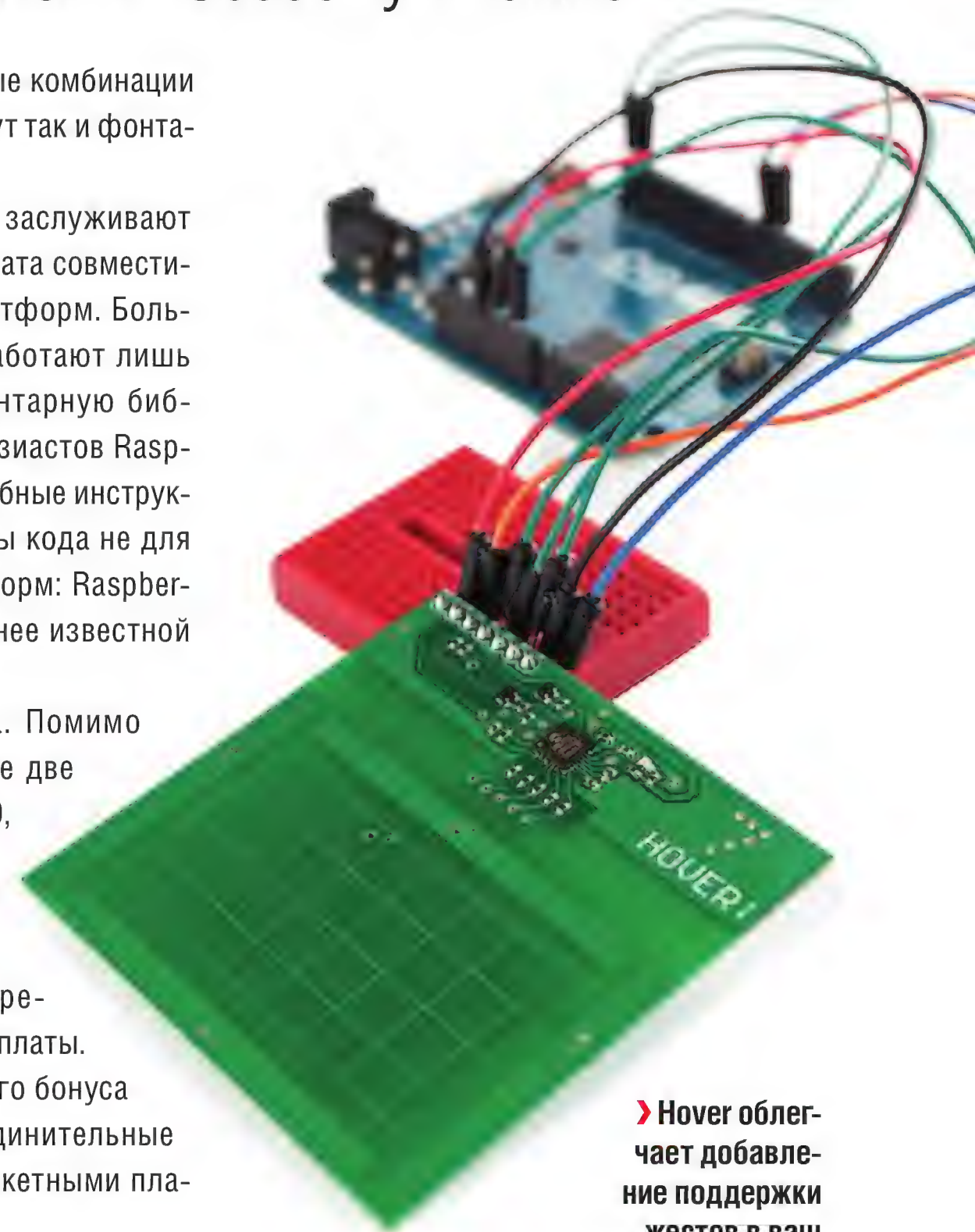
Установка тоже проста. Помимо соединения I2C, есть также две соединительные шины GPIO, причём плата поддерживает и 5-вольтовые, и 3,5-вольтовые микроконтроллеры с помощью переключения на уровне логики платы. В качестве редкого приятного бонуса к Hover уже припаяны соединительные клеммы, совместимые с макетными платами [breadboard].

Лёгкая настройка

Мы протестировали Hover с Raspberry Pi и отметили, что он очень прост в настройке. Если вы уже знакомы с соединениями I2C, то дело лишь за установкой макетной платы и скачиванием предлагаемой библиотеки для Python. Будучи довольно элементарной, эта библиотека имеет лучшую документацию, виденную нами за последнее время, и она продумана так, чтобы помочь хакерам и мастерам всех уровней выжать из платы максимум.

Пример скрипта для Hover даёт возможность быстро перенести в ваш проект код, совместимый с Hover, но команде было бы неплохо обновить библиотеку, чтобы она поддерживала многопальцевые касания «из коробки» — по крайней мере, на момент написания данной статьи библиотека не обновлялась уже пять месяцев.

На официальном сайте Hover Labs (www.hoverlabs.co/projects) можно найти массу отличных примеров и идей для проектов, в том числе раздел, посвящённый управлению старыми играми. Но хотя нам и нравится идея посылать Фроггера [Frogger] под колёса машин лёгким движением руки, самое большое впечатление на нас произвело видео, которое показывает



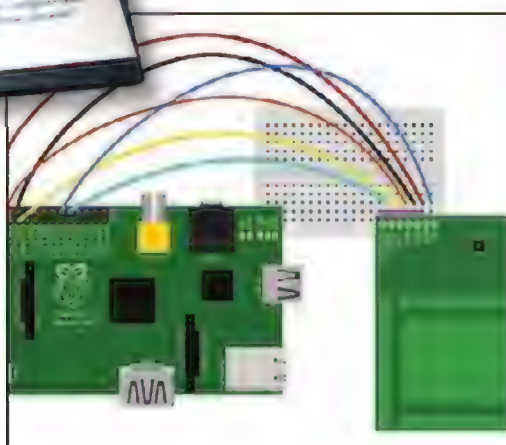
» Hover облегчает добавление поддержки жестов в ваш аппаратный проект.

простую реализацию управления Google Earth комбинацией касаний и жестов.

Также Hover довольно легко использовать с каким угодно компьютером и приложением. С помощью Arduino Leonardo или подобного компьютера вы можете подсоединить Hover как псевдоустройство ввода, заставив почти любой компьютер воспринимать его как обычную клавиатуру или мышь. Умная штука. **LXF**

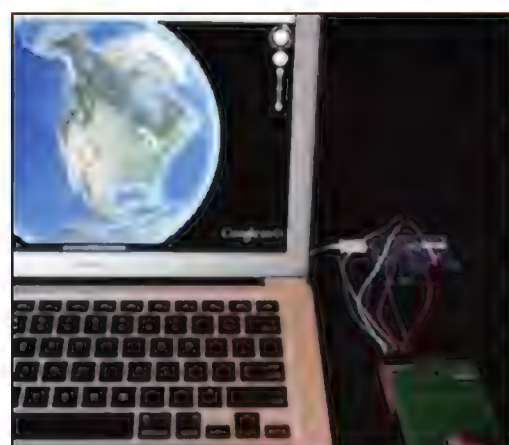


Свойства навскидку



Простая установка

Настройка через I2C хорошо описана на сайте для всех четырёх совместимых устройств.



Устройство ввода

Используйте Arduino Leonardo, и большинство систем сочтут Hover просто USB-мышью.

LINUX FORMAT Вердикт

Hover

Разработчик: Hover Labs
Сайт: www.hoverlabs.co
Цена: £32

Функциональность	8/10
Производительность	8/10
Удобство в работе	9/10
Оправданность цены	9/10

» Hover невероятно упрощает добавление управления жестами почти в любой мыслимый проект.

Рейтинг **9/10**

Google Nexus 6

Мэтт Свайдер опробовал фаблет от Google, подслащённый Android Lollipop.

Вкратце

» Первый телефон, получивший Android 5.0, заодно может похвастаться лучшими характеристиками и самым большим экраном, который вы видели.

Спецификации

- » Процессор: Qualcomm Snapdragon 805
- » Графика: Adreno 420
- » ОЗУ: 3 ГБ
- » Дисплей: 5,96-дюймовый QHD AMOLED, 2560×1440
- » Память: 32 или 64 ГБ
- » Разъёмы: Micro USB 2.0, Nano-SIM, аудио 3,5 мм
- » Связь: 802.11ac 2×2 (MIMO), Bluetooth 4.1, NFC
- » Камеры: тыльная 13 МПикс, фронтальная 2 МПикс
- » Батарея: 3220 мА·ч
- » Габариты: 82,98×159,26×10,06 мм
- » Вес: 184 г

» Металлическая рамка — грандиозное улучшение по сравнению с Nexus 5.

Nexus 6 стал первым смартфоном с Android 5.0 Lollipop, и нет лучшего способа ощутить все изменения, которые Google реализовал в своей ОС. Он демонстрирует новый интерфейс — Material Design — на большом 6-дюймовом экране, с лёгкостью переключается между приложениями благодаря последнему процессору Snapdragon и 3 ГБ ОЗУ, и выдерживает целый день с аккумулятором на 3220 мА·ч, поддерживающим беспроводную подзарядку.

Nexus 6, конкурентам на зависть, имеет 5,96-дюймовый дисплей AMOLED, который не только больше и ярче Moto X, но и по всем показателям превосходит 5,5-дюймовый iPhone 6 Plus, ведь его разрешение вдвое больше, чем Full HD у Apple, и составляет 2560×1440 точек, или 493 точки на дюйм.

Уже есть и другие члены клуба «Quad HD», в том числе и на Android, такие как LG G3 и Samsung Galaxy Note 4, но Google Nexus 6 выделяется среди них своим гигантским экраном.

Дисплей простирается до самых краёв корпуса, и софт-клавиши, расположенные на нём же, увеличивают его площадь, одновременно уменьшая габариты всего телефона.

При своем размере Google Nexus 6 хорошо лежит в руке. Дизайн от Motorola, позаимствованный у Moto X, представляет собой прочную алюминиевую рамку и плавно изогнутую заднюю сторону. Изгиб выглядит естественным для руки — или, в большинстве случаев, для рук — а пластиковая задняя часть ощущается по меньшей мере гладкой. Nexus 6



» Nexus 6 кажется лучшим фаблетом из нами виденных.

доступен только в «облачном белом» или «ночном синем» цветах, зато есть преимущество в виде поддержки беспроводной подзарядки Qi, которой не могут похвастаться телефоны Moto X.

Новый телефон от Google также выгодно отличается двумя динамиками вверх и вниз и поставляется вместе с зарядным устройством Turbo Charger, которое обеспечивает телефону 6 часов работы всего за 15 минут.

Что внутри?

В сердце Nexus 6 находится четырёхъядерный 2,7-ГГц процессор Qualcomm Snapdragon 805 с графическим чипом Adreno 420. Это самый мощный чип для самого большого на сегодня Android-смартфона от Google. Чтобы не ронять достоинство смартфона, предусмотрено 3 ГБ оперативной памяти и разумные 32 ГБ либо 64 ГБ встроенного хранилища: никаких дурацких 16 ГБ, порочащих идею.

Все эти характеристики Nexus 6 — не излишества: они важны для лёгкого, но богатого на возможности Android Lollipop, который предустановлен в телефоне. Android научился делать в фоне больше действий, чем когда-либо: держать запущенными много приложений, получать уведомления, сразу попадающие на экран блокировки, что делает Google Now мгновенно доступным по движению пальца влево, и выполнять голосовой поиск, когда вы произносите злополучную фразу «OK, Google».

Сзади имеется 13-мегапиксельная камера — можно подумать, что такая же,

как и у Moto X с 13 мегапикселями. Но это не так. Камера оснащена лучшим сенсором Sony IMX214 с более широкой диафрагмой f/2.0 и оптической стабилизацией изображения.

В США Nexus 6 открылся для продажи 12 ноября прошлого года, и мгновенно разошёлся по предварительным заказам. Nexus 6 должен был стать доступным на Google Play в течение того же ноября, но этого не произошло. Однако, если верить прессе, Nexus 6 был выпущен оператором O2 по цене £479 за модель с 32 ГБ: вполне справедливая цена для предлагаемых характеристик и объёма памяти. На Google Play Nexus 6 стоит £499. По такой же цене продается 4,7-дюймовый iPhone 6 с 16 ГБ, уступающий Nexus 6 и физически меньшим экраном, и по объёму памяти. **LXF**

LINUX FORMAT Вердикт

Google Nexus 6

Производитель: Motorola
Сайт: www.google.com/nexus/6
Цена: £499

Функциональность	10/10
Производительность	10/10
Удобство в работе	8/10
Оправданность цены	9/10

» Гигантский дисплей, топовые характеристики, а также дебют Android 5.0 делают Nexus 6 лучшим на сегодня фаблетом.

Рейтинг **9/10**

Google Nexus 9

Google поступился принципом низких цен ради входа в верхний сегмент рынка планшетов. **Мэтт Свайдер** задумался: сбегут ли пользователи Apple на Android?

Спецификации

- » Процессор: 64-битный 2,3-ГГц Nvidia Tegra K1
- » Графика: Kepler DX1
- » ОЗУ: 2 ГБ
- » Дисплей: 8,9-дюймовый IPS LCD, 2048×1536, Corning Gorilla Glass 3
- » Память: 16 или 32 ГБ
- » Разъёмы: Micro USB 2.0, аудио 3,5 мм
- » Связь: 802.11ac, Bluetooth 4.1, NFC
- » Камеры: тыльная 8 МПикс, фронтальная 1,6 МПикс
- » Батарея: 6700 мА·ч
- » Габариты: 154×228×8 мм
- » Вес: 425 г (Wi-Fi), 436 г (LTE)

Nexus 9 от Google был спроектирован HTC, чтобы стать идеалом среди планшетов с чистым Android, и в основном справляется с задачей быть «вполне достойным» в сравнении с чем угодно, кроме iPad. Он не такой большой и отсталый, как Nexus 10 производства Samsung, и не такой маленький, как Nexus 7, сделанный ASUS. Это серебряная пуля среди планшетов — но он будет стоить ваших трудовых дензнаков.

Среди его соперников на Android — Samsung Galaxy Tab S, который противопоставляет 8,9-дюймовому планшету Google свои 10,5 либо 8,4 дюйма, и Sony Z3 Tablet Compact. Но, помимо «железа», за Nexus 9 говорит тот факт, что это пока единственный шанс испытать Android 5.0 Lollipop на планшете. Это делает его достаточно привлекательным продуктом с разных точек зрения. Новый единый стиль «Material Design» делает яркими и насыщенными как операционную систему, так и собственные приложения Google.

Если вспомнить, что он сделан той же компанией, которая дала нам HTC One M8 с его лощёной внешностью, то неудивительно, что Nexus 9 имеет обод из матового металла, но мягкую обрезиненную заднюю крышку — такую же, как на более компактном Nexus 7.

8,9-дюймовый LCD-дисплей IPS у Nexus 9 немного меньше, чем 9,7-дюймовый экран iPad Air 2, но предлагает такое же максимальное разрешение.

Физические кнопки на ободке не всегда легко найти — если вы привыкли к Nexus 6, то предпочли бы более заметную



» Первый планшет с Android 5.0 Lollipop — уверенное, если не впечатляющее предложение.

кнопку включения. К счастью, разыскивать эту крохотную кнопочку не всегда обязательно, если планшет лежит на столе: можно воспользоваться удобной новой функцией «проснуться по двойному касанию» в Android 5.0 Lollipop.

Самое крупное упущение — это отсутствие слота для microSD. Никакого расширения хранилища не предусмотрено, а значит, модель с 16 ГБ становится сомнительным выбором, если вы хотя бы немного используете непотоковое мультимедиа.

Производительность

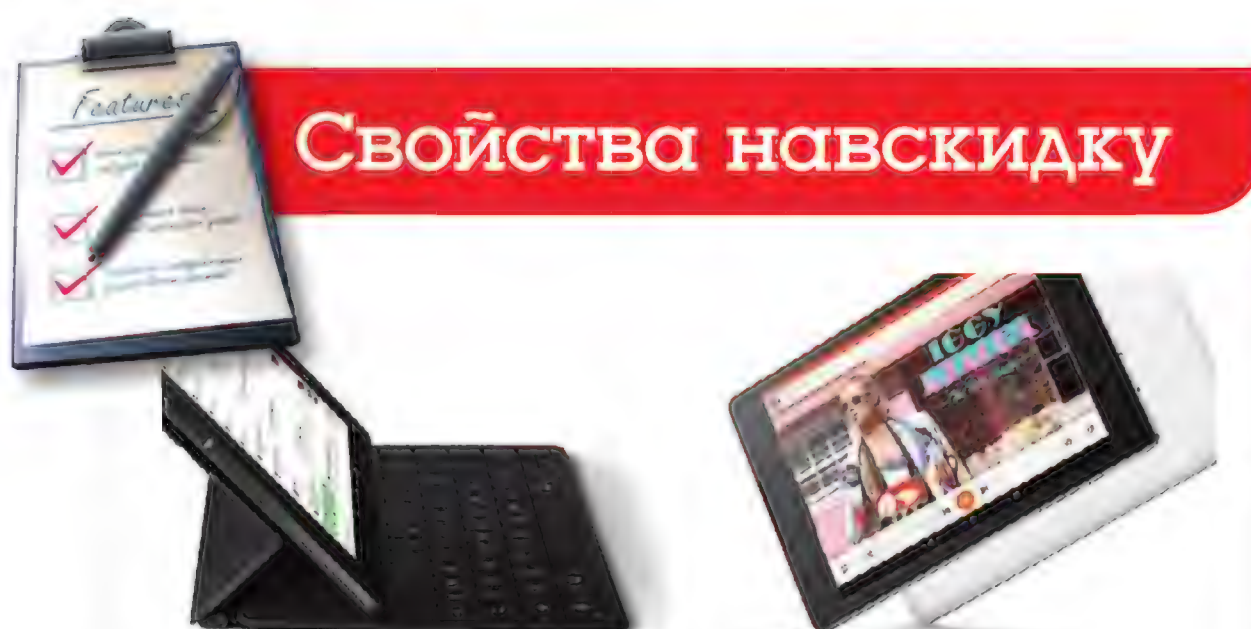
Процессор Nvidia K1 Tegra пришёл на смену привычному чипсету Qualcomm Snapdragon, который скрывался за дисплеями Android-планшетов ранее, но хорошая новость в том, что это по-прежнему 64-битная система. Новый Android Lollipop использует преимущества 64-битной архитектуры, что даёт Nexus 9 задел на будущее. Вместе процессор и Lollipop должны дать хорошие результаты в мощных и полезных приложениях.

Тест Geekbench 3 на одном ядре на Tegra K1 превзошёл таковой на новом iPad, составив в среднем 1939, тогда как один процессор iPad показал среднее значение 1815 в похожих тестах — хотя, конечно, Tegra K1 — это двухъядерный процессор, а не трёхъядерный, как в iPad.

Наш тест аккумулятора на полной яркости показал, что 90-минутное Full HD

видео снизило заряд батареи до 82 % — это уровень, до которого iPad Air 2 просто не дотянуть. В других тестах из реальной жизни Nexus 9 продержался без подзарядки полтора дня. Время жизни аккумулятора на планшете не так важно, как на телефоне, но Nexus 9 отнюдь не слабак.

Если вы выбираете планшет с голым Android, то Nexus 9 годится на эту роль и имеет ряд реальных преимуществ, выводящих его в лидеры. Впрочем, он не является победителем ни в одной категории, и будет интересно посмотреть, как Google станет использовать эту базовую модель, чтобы со временем улучшить всю экосистему планшетов. **LXF**



Свойства навскидку

Клавиши для удобства

Обложка с клавиатурой (продаётся отдельно) соединяется по Bluetooth и использует NFC для простого подключения.

QHD-дисплей

Разрешение высокое, хотя цветопередача и контрастность бесспорно средние. Фильмы в HD не слишком впечатляют.

LINUX FORMAT Вердикт

Google Nexus 9

Производитель: HTC
Сайт: www.google.com/nexus/9
Цена: £319

Функциональность	8/10
Производительность	9/10
Удобство в работе	8/10
Оправданность цены	7/10

» Обновление для прошлогоднего Nexus 7 стало отличным дебютом для Lollipop, но не сравнялось с iPad Air 2.

Рейтинг 8/10

Chromebook 2 от Samsung

Samsung отверг тренд Chromebook'ов и создал необычайно приятный и тонкий безвентиляторный ноутбук, опередив конкурентов, как считает **Кевин Ли**.

Вкратце

» Samsung Chromebook 2 — первая попытка показать будущее хромбуков, причём очень яркая.

Спецификация

- » CPU: Intel Celeron N2840 2,58 ГГц
- » Графика: Intel HD Graphics
- » ОЗУ: 2 ГБ DDR3L
- » Дисплей: 11,6 дюймов, 1366×768
- » Дисковая память: 16 ГБ SSD
- » Порты: 1×USB 3.0, 1×USB 2.0, HDMI, разъём для Micro SD, разъём для наушников/микрофона
- » Связь: 802.11ac Wi-Fi, Bluetooth 4.0
- » Камера: веб-камера HD 720p
- » Батарея: 4080 мА·ч
- » Вес: 1,2 кг
- » Габариты: 290×204×17 мм

Объявление Samsung о выходе своего второго Chromebook заставило нас обернуться. Вместо подражания остальным хромбукам с дешёвыми пластиковыми корпусами Samsung выдал своему новому облачному ноутбуку крышку с отделкой под кожу, а для 13-дюймовой модели ещё и дисплей высотой в 1080 пикселей. К сожалению, Samsung снабдил его своим ARM-процессором Exynos, предназначенным для планшетов, поэтому в условиях многозадачности он буксует.

Спустя пять месяцев Samsung обновил 11,6-дюймовый вариант Chromebook 2, заменив мобильный процессор на Intel Celeron. Посмотрим, превратит ли эта простая замена Samsung Chromebook 2 в лидера, каким он хочет быть.

Chromebook 2, кажется, собран даже лучше, чем многие ноутбуки среднего класса. Крышка по бокам отделана кожзаменителем, что напоминает дизайн Samsung Galaxy S5, вплоть до искусственных швов.

Остальные части корпуса выполнены из светло-серого пластика, но не прогибаются, даже если давить на клавиатуру, в основном благодаря внутреннему металлическому каркасу, скрепляющему весь корпус. Углы также усилены металлом, что даёт дополнительную защиту в случае падения ноутбука.

Вдобавок перед нами один из самых тонких хромбуков на сегодня: его размеры — 290×204×17 мм. Весит большинство хромбуков около 1,3 кг, но Samsung Chromebook 2 обошёл большинство из них, при весе 1,2 кг.

Характеристики покажутся вам знакомыми, если вы сравните их с другими вариантами на базе Chrome OS, такими как Dell Chromebook 11 или Acer C720. У Samsung очень похожие данные: 11,6-дюймовый дисплей разрешением 1366×768, 16 ГБ SSD и 2 ГБ оперативной памяти. Но, несмотря на сходство, у Samsung есть козырь: новый 2,58-ГГц процессор Intel Celeron N2840. Это не просто замена чипу Exynos, а представитель технологии Bay Trail от Intel, предназначенной для систем без охлаждения.



» Плавные обводы Samsung Chromebook 2 должны привлечь к Chrome OS много новых пользователей.

Другие хромбуки, включая Acer и Dell, чаще всего используют Celeron 2955U на 1,4 ГГц, из семейства Intel Haswell. Чип Bay Trail нагревается чуть меньше, не заставляя вас слушать шум вентилятора, но жертвует производительностью, становясь немного более медленным.

Задержки остались

Также мы обнаружили заметные задержки в работе *Chrome*. Как ни странно, заикания происходили, когда мы пытались набрать адрес или поисковый запрос в адресной строке. Чип от Intel также иногда тормозил на документах на Диске Google, особенно на документах с таблицами и графиками. Зато Chromebook 2 без труда открывал 25 вкладок в браузере, заодно воспроизводя музыку в Google Music. Также чип Bay Trail не выказал задержек при просмотре 1080p-видео на YouTube.

Другая беда хромбуков в том, что почти все они оснащены гнусными линиями дисплеями, за редкими исключениями вроде Chromebook Pixel. Chromebook 2 удалось — хоть и не вполне — сломать и этот тренд, применив яркий дисплей, хоть и не идеальный, но хороший. Цвета не совсем живые, особенно учитывая склонность Samsung к перенасыщенным

панелям AMOLED, но точны. Более того, вы действительно увидите чёрный таким, каким он должен быть, а не серые пятна.

Аккумулятор в наших тестах продержался 7 часов 32 минуты — отлично, но маловато, чтобы тягаться с Acer C720 с его невероятными 8 часами 30 минутами работы.

В целом, новый Samsung Chromebook 2 лучше оригинала. Хотя и есть придирки к производительности, он очевидно идёт впереди толпы. **LXF**

LINUX FORMAT Вердикт

Samsung Chromebook 2

Разработчик: Samsung
Сайт: www.samsung.com
Цена: \$249,99

Функциональность	9/10
Производительность	6/10
Удобство в работе	9/10
Оправданность цены	10/10

» Samsung Chromebook 2 — лишь первая ласточка, предвещающая возможное яркое будущее всех Chromebook.

Рейтинг **8/10**

Deponia

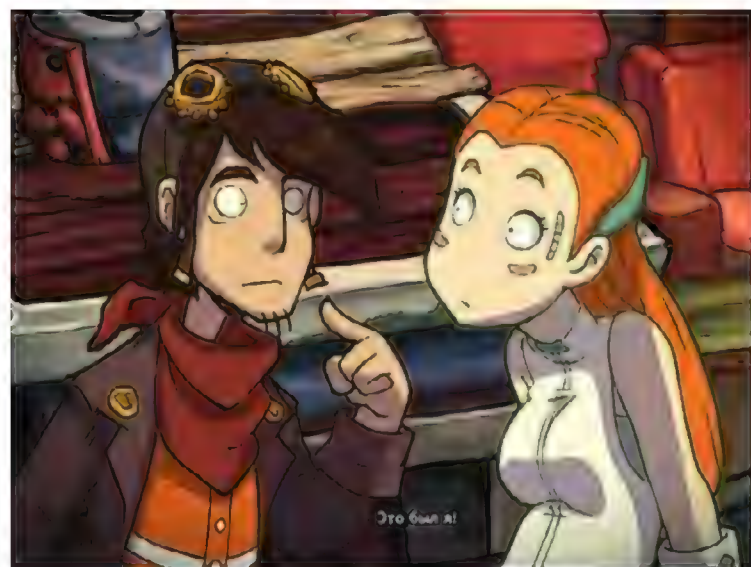
Евгений Балдин оторвался от прохождения второй части трилогии, дабы записать впечатления о первой.

Вкратце

» Не совсем классический point-and-click-квест в мире не совсем классической утопии.

В 2007 году собрались двое друзей, взяли дипломную работу одного из них и написали квест... Так возникла немецкая фирма Daedalic Entertainment, которая с тех пор регулярно выпускает, может быть, и не прямо-таки шедевры всех времён и народов, но вполне интересные и заслуживающие изучения истории для любителей развлечений в стиле «указал и нажал».

В 2012 году возникла история *Deponia*. Это мир-утопия. Правда, начинается повествование с нижнего конца этой утопии, то есть оттуда, куда жители утопии скидывают свой мусор. И среди этого мусора тоже живут люди. Вполне себе нормальные, если учесть влияние на них окружения. Не сказать, что они особо довольны такой жизнью, но в основном смирились с диковатым окружением и как-то к этой жизни притёрлись. Рисовка довольно приятная, хоть и несколько схематичная. В частности, почему-то у молодых женщин не видно носа. Видимо, это их отличительный признак.



» Это, очевидно, была любовь с первого взгляда. Не совсем, конечно, но история именно об этом.

Свойства навскидку



Негероический герой

Неглупый и в тоже время забавный квест с наглым и эгоистичным главным героем. Наш Руфус ухитрился смастерить «ракетный трактор»...



Интересный мир

Интересны и персонажи, его населяющие. Не всегда очевидные, но логичные загадки. Чего не отнять у мусорной планеты, так это красивые закаты.



» Берлога ГГ на мусорной планете. По телевизору тоже один мусор.

Но главный герой Руфус — не такой: он мечтает сбежать с этой помойки его жизни. Благо есть куда — на Элизим; но есть и проблема: этот остров всеобщего счастья находится далеко в небе. Иначе как бы его жители смогли так легко гадить на Депонию, причём даже не подзревая, что там кто-то живёт? Руфус далёк от классического героя классического квеста настолько, насколько возможно. Это эгоистичный, самовлюблённый социальный паразит, считающий, что все ему должны и обязаны. Герой по жизни в целом довольно безобиден, но уж очень активен и беспринципен. Что может изменить такое существо? Конечно, настоящая любовь! Не сразу, не полностью, но всё-таки Руфус осознаёт, что есть черта, которую он не переступит, и есть та, кого он не предаст. Но до этого осознания вам идти и идти по игре вместе с ним, смеяться над неожиданными вывертами сюжета и наслаждаться тем, как «вселенский уравниватель» воздаёт хаму за его нехорошее поведение. Вот бы и в жизни так. Это скорее мир парпанка, несмотря на наличие ракетной техники и электричества.

Сам квест не очень продолжителен: по времени — как книгу средних размеров прочитать, или 8 часов на неспешное путешествие с изучением всех активных точек и описаний с тремя-четырьмя заглядываниями в инструкцию по прохождению в Интернете. Русские субтитры вполне адекватны. Мини-игры, к счастью, можно пропускать. Это не влияет ни на что, кроме «достижений». Помимо самой *Deponia*, в свет вышли два её продолжения: *Chaos*

on *Deponia* и *Goodbye Deponia*. То есть ещё два раза по столько объёма удовольствия обеспечено.

Следует отметить, что не сразу эти истории стали доступны в окружении GNU/Linux. Только летом 2014 года трилогия была портирована на GNU/Linux. Связано это с тем, что игровой движок Visionaire Adventure Game Engine от сторонней компании тоже только этим летом стал поддерживать сборку Linux-бинарников. К сожалению, порт доступен только через сервис Steam, хотя сама *Deponia* распространяется и через DRM-free сервис gog. Но я надеюсь, что в будущем игры Daedalic Entertainment уже не будут так долго добираться до благодарной аудитории сообщества, а сразу порадуют нас своей непосредственностью и добротой. **LXF**

LINUX FORMAT Вердикт

Deponia

Разработчик: Daedalic Entertainment
Сайт: daedalic.ru
Цена: 699 р. на Steam за трилогию *Deponia: The Complete Journey*

Сюжет	8/10
Графика	7/10
Увлекательность	8/10
Оправданность цены	8/10

» Тем, кому нравятся квесты, это произведение игрового искусства проходить обязательно. Остальным квесты могут начать нравиться сразу после.

Рейтинг 8/10

Football Manager

Том Хэтфилд открывает сезон FM2015 этой пресс-конференцией.

Вкратце

» Последнее ежегодное обновление двадцатилетней франшизы. Основной игровой процесс всё тот же, но перемены восторга не вызывают.

» Прыжки к штанге. Красивая игра. Мужики, гоняющие мяч по полю.

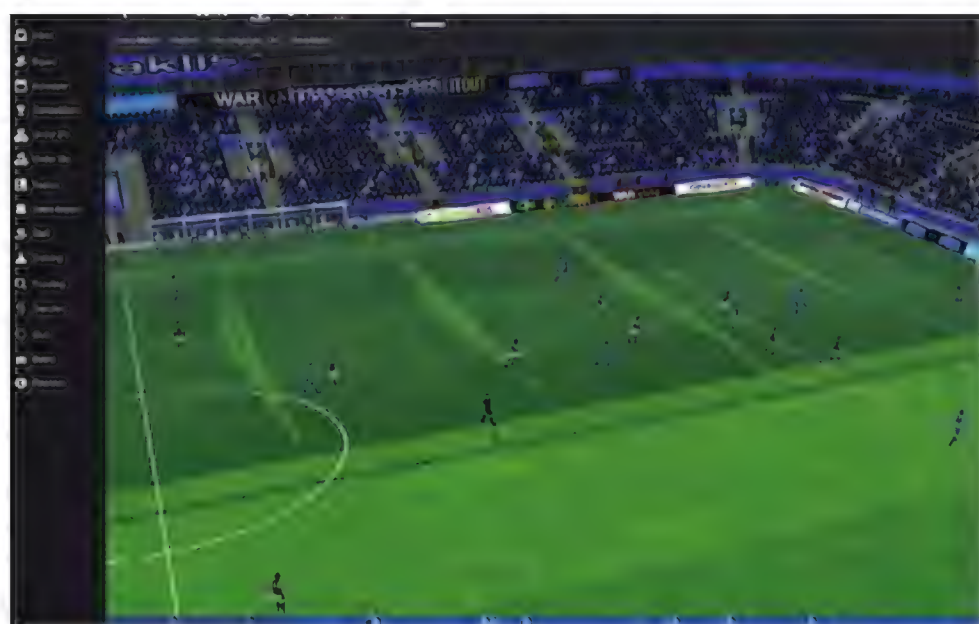
Игру *Football Manager* часто считают «футбольной ролевкой» (RPG): вы окунаетесь в жизнь футбольного менеджера и в стратегию игры вокруг неё. Такие идеи фантастически звучат на стадии планирования, но каждая попытка Sports Interactive воплотить её заканчивается лишь проведением очередной пресс-конференции. Бич *Football Manager 2015* — слишком долгая разминка. Охота за игроками идёт в несколько этапов, а их способности изображены

как ряд навыков, который вы постепенно сужаете. Но за это игру осуждать трудно — родной Классический режим ценит ваше время и позволяет быстро перейти к игре. Интерфейс пользователя улучшен, и не только внешне: он также воплотил мудрые решения.

Проблема посерьёзнее — движок матча. Критика тут рискованна: его работа столь восхитительно непрозрачна, что игроки давно привыкли воспринимать её как силу природы. В футболе *FM2014* были прекрасные пасы игроков; в *FM2015* они играют, как пьяный сброд. Эстетика в футболе, может, и не главное, но после пары часов в *FM2015* мы поняли, как нам не хватает изысканных навыков нашей старой команды.

Чтобы наш отзыв не прозвучал чересчур уж негативно, подчеркнём: основа игры всё та же и работает

не хуже. Не исключено, что традиционный патч в середине сезона полностью сменит тон движка матча — тогда мы от всей души его порекомендуем. Но на данный момент *Football Manager 2015* — это год насмарку. **LXF**



LINUX
FORMAT

Вердикт

Football Manager 2015

Разработчик: Sports Interactive
Сайт: www.footballmanager.com
Цена: £35

Увлекательность	6/10
Графика	6/10
Реиграбельность	9/10
Оправданность цены	7/10

» Медленно и удручающе старо: не звезда серии. Если у вас есть FM2014, смело проходите мимо.

Рейтинг **7/10**

Приключения в MC

Джонни Бидвелл пришёл в мир *Minecraft* ради удовольствия от кодинга.

Вкратце

» Коллекция проектов с применением API Python для *Minecraft*, приготовленных специально для детей. Но это неплохое введение в программирование и для всех остальных.

» Стив, главный герой *Minecraft*, отлично знает, который нынче час.

От сайта Мартина О'Хэнлона [Martin O'Hanlon] про *Minecraft* нелегко оторваться (посетите <http://stuffaboutcode.com>). На основе этого сайта Мартин вместе с Дэвидом Уэйлом [David Whale] выпустили книгу «Приключения в *Minecraft* [Adventures in *Minecraft*]», цель которой — объяснить API Python для *Minecraft* детям. Текст разделён на 9 проектов, сложность которых возрастает — от обязательного Hello *Minecraft* World до применения Pi или Arduino и GPIO-компонентов для создания игры с рискованным



сбором кристаллов, где вы соревнуетесь на время с часами (7-сегментными светодиодами), созданными на техниках, изученных в предыдущих частях.

Текст независим от оборудования и объясняет, как настроить *Minecraft* на Pi, PC или Mac с помощью сервера Bukkit и плагина RaspberryJuice, который можно загрузить с сайта-партнёра книги. Книгу поразительно легко читать, что делает её хорошим учебным пособием. Разнообразные врезки содержат всякие подробности, объяснения, предупреждения, а также наблюдения авторов, которые обозначены мультяшными аватарами.

Код по возможности придерживается чистоты и минимализма, с понятными объяснениями там, где они необходимы. Даже в большом финальном проекте, описывающем такую довольно сложную материю, как потоки, код разбит на простые для усвоения фрагменты.

С небольшим понуканием, даже самые непоседы сумеют что-то почерпнуть из этой книги. Книга делает немаловажную работу по объяснению фундаментальных

основ программирования и заодно намекает на возможность дальнейших приключений, которые воодушевят юных фантазёров. **LXF**

LINUX
FORMAT

Вердикт

Adventures in Minecraft

Авторы: Martin O'Hanlon, David Whale
Издатель: John Wiley & Sons
ISBN: 978-1-118-94691-6
Сайт: www.wiley.com/go/adventuresinminecraft
Цена: £15
Страниц: 320

» Увлекательнейшая книжка для класса или для дождливых выходных. Взрослым читателям изложение может показаться слишком неспешным.

Рейтинг **9/10**

Red Hat Enterprise Linux

предоставляет вам производительность,
масштабируемость, безопасность и надежность,
ранее доступные только на очень дорогих платформах

Самая популярная в мире
Linux платформа для бизнеса

Обеспечивает высокую
производительность, надежность,
масштабируемость и безопасность

Сертифицирована ведущими
производителями оборудования
и разработчиками ПО



Совместима с широким спектром
оборудования от рабочих станций
до серверов и мэйнфреймов

Обеспечивает одинаковые
условия работы приложений
при использовании в физической,
виртуальной и облачной средах

Пользователи RHEL экономят на оборудовании,
лицензиях на программное обеспечение
и эксплуатационных расходах



ГНУ/Линуксцентр — Linux-эксперт для вашего бизнеса

- Premier Business Partner компании Red Hat
- 12 специалистов по разработке и внедрению, сертифицированных компанией Red Hat
- Более 100 клиентов, использующих Red Hat
- 10 лет на рынке

Red Hat — ведущий серверный дистрибутив Linux

- Более 15 лет промышленного использования
- Свыше 80% рынка корпоративного Linux по данным CIO Insight
- 5 лет среди лучших вендоров
- Выгодная совокупная стоимость владения (TCO)
- Поддержка в течение 10 лет

Специальное предложение!

Закажите Red Hat Enterprise Linux в ГНУ/Линуксцентре и получите
в подарок книгу «Полное руководство пользователя Red Hat Enterprise Linux»



Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center



НОВОСТИ Adobe — и на Android » Google расторгнулся » Офисы в телефоне
» 3D-печать с телефона

РАЗРАБОТКИ

Adobe заметила Android

Компания Adobe выпустила для Android приложение *Adobe Lightroom Mobile*.

Теперь редактирование своих изображений, хранящихся в Creative Cloud, доступно и пользователям мобильной ОС от Google.

Через Play Store версия *Lightroom* для Android распространяется бесплатно, однако необходима подписка на облачный сервис Adobe. Для оценки возможностей Creative Cloud предлагается 30-дневный пробный период; в дальнейшем стоимость подписки составит \$10 в месяц.

Любопытно, что представленное приложение в настоящее время работает только на смартфонах, а версия для планшетов находится пока в разработке. Когда ранее Adobe создавала аналогичное приложение для iPhone и iPad, ситуация была обратной: первоначально была обеспечена поддержка планшетов Apple.

Интерфейс версии для Android практически идентичен аналогу для iOS, что положительно оценят те пользователи, которым в течение дня приходится сталкиваться с разными ОС.

Возможности редактирования ограничены самыми основными инструментами — как и следовало ожидать, по функциональности мобильный вариант *Lightroom* сильно уступает полной версии. Но, с учётом размера экранов мобильных устройств, это решение следует признать правильным: более сложное редактирование в любом случае удобнее и легче выполнять на ноутбуке или настольном ПК.

Lightroom Mobile для Android предлагает:

- » автоматическую синхронизацию всех правок *Lightroom Mobile* и *Lightroom* для настольных ПК;
- » удобную публикацию фотогалерей в Интернете;
- » поддержку самых разнообразных форматов изображений;
- » просмотр отзывов о галереях с общим доступом — отзывы также синхронизируются с коллекциями *Lightroom* на настольном ПК;
- » автоматический импорт фотографий с мобильного устройства сразу после съемки, а также воз-

можность самостоятельно добавлять файлы в синхронизированную коллекцию;

» более 40 встроенных стилей, позволяющих преобразовывать изображения одним щелчком мыши (цветовые эффекты, настройка четкости изображения, уменьшение шума, эффекты виньетирования и множество других возможностей).



» Все правки *Lightroom Mobile* автоматически синхронизируются с *Lightroom* для настольных ПК.

МАГАЗИНЫ

И Google Play впереди

Ассортимент Google Play превзошёл Apple App Store.

Компания appFigures представила отчёт о динамике сетевых хранилищ приложений в 2014 г. Наиболее значимым событием в этой области следует признать тот факт, что впервые по числу новых приложений сетевой магазин Google Play превзошел Apple App Store. Google закончил год, имея 1,43 млн приложений, по сравнению с 1,21 млн у Apple (для сравнения, каталог Amazon увеличился на 90%, однако общее количество приложений в нём всего около 293 тыс.). Фрагментация рынка и отличающиеся возможности устройств не снижают энтузиазм разработчиков для Android. Перспективные Apple Watch, Swift и большой экран служат хорошим стимулом также и разработчикам для Apple.

В течение 2014 г. количество приложений в Google Play более чем удвоилось. Впрочем, для всех трёх магазинов приложений рост составил не менее 50% (относительно конца 2013 г.).

Для iOS в 2014 году в App Store добавлено более 128 тыс. новых бизнес-приложений. Высокий рейтинг категорий Business или Games был достаточно предсказуем, но второе место по темпу роста у категории Food & Drink вызывает некоторое удивление. На Google Play более чем вдвое вырос раздел Games. Интересно, что крошечная по сравнению с Games категория Photography пережила в 2014 г. аномальный рост — вероятно, в связи с всплеском популярности жанра селфи. Далее следуют Music, Business и Entertainment.

Третий год подряд сообщество разработчиков Google растёт быстрее, чем у Apple. На Google Play распространяются приложения от 388 тыс. разработчиков — больше, чем у Apple (282 тыс.) и Amazon (48 тыс.), вместе взятых. Причём у Apple большинство новых разработчиков помещают в магазин бизнес-приложения, а у Google они сосредоточены на играх — самой быстрорастущей

категории Google Play. С 2008 г., когда появились App Store и Google Play (вначале — Android Market), компания Apple привлекала потребителей большим числом приложений для своих устройств.



» Впервые по числу новых приложений Google Play превзошел Apple App Store.

ПАКЕТЫ

LibreOffice на Android

Свободный офисный пакет идёт в телефоны.

В середине января The Document Foundation и британская компания Collabora представили бета-версию *LibreOffice Viewer* для Android. Данное приложение, первое в своем роде, обеспечивает просмотр текстовых документов, простых презентаций (поддержку сложных презентаций планируется реализовать в дальнейшем) и электронных таблиц формата ODT, OPD, ODS, OTT, OTP, DOCX, PPTX, XLSX, DOTX, XLTX, PPSX, DOC, PPT, XLS, DOT, XLT, PPS, т.е. созданных в *LibreOffice*, *OpenOffice*, *Microsoft Office 2013/2010/2007/2003/XP/2000/97*. Таким образом, возможности приложения перекрывают потребности большинства пользователей — за исключением работающих с чем-то совсем уж экзотическим, типа *AbiWord*.

LibreOffice Viewer для Android на Google Play добился рейтинга 4,5 звезды, что очень неплохо для первого бета-релиза. В приложении задействован тот же движок, что и в *LibreOffice* для настольных ПК, а также новый интерфейс, базирующийся на *Firefox* для Android.

Скептики напоминают, что мобильная версия *Microsoft Office* давно доступна для сторонних

платформ, в том числе и для Android, и при этом обладает более богатой функциональностью — она, помимо прочего, обеспечивает возможности редактирования документов. Предварительные версии *Word*, *Excel* и *PowerPoint* для планшетов на Android предлагаются для загрузки в магазине Google Play (из этих трех приложений самым популярным оказался *Word*: от 100 тыс. до 500 тыс. установок). Однако даже скептикам невозможно не признать, что развитие *LibreOffice* идёт в правильном направлении. Директор TDF Бьорн Михаэльсен [Björn Michaelsen] назвал этот релиз «первым в новой серии мобильных приложений» (т.е. мы можем надеяться на дальнейшие сюрпризы) и призвал частных лиц, компании и организации принять участие в процессе разработки, присоединившись к сообществу *LibreOffice*.

Чуть позднее стало известно, что Collabora и испанская компания Igalia выиграли объявленный TDF в сентябре прошлого года тендер на создание полнофункциональной версии *LibreOffice* для Android. Ожидается, что представление первой бета-версии этого продукта состоится в марте, однако о предусмотренной в ней функциональности,



➤ Правильной дорогой идёте, товарищ *LibreOffice*.

как и о конкретных сроках выхода финальной версии, пока что никакой информации не поступало.

Здесь же будет к месту упомянуть о факте приобретения сервисом Dropbox, с целью расширения офисных возможностей своих приложений, израильской компании CloudOn.

В любом случае, повышение интереса разработчиков к офисным приложениям для мобильной операционной системы от Google выгодно пользователям, поскольку конкуренция улучшает качество предлагаемых продуктов.

ОБОРУДОВАНИЕ

Android App и трехмерка

Освоена печать 3D-объектов из приложения для Android.

Американская компания Makerbot — производитель наиболее популярных 3D-принтеров, поддерживающих Linux. В настоящее время компания предлагает приложение для устройств на Android, дающее пользователям возможность управлять 3D-печатью с помощью смартфона. Это второй программный продукт для Android от Makerbot; предыдущий — Thingiverse — более ориентирован на обмен идеями между участниками сообщества Makerbot. Makerbot Mobile 1.0 для Android совместим с моделями принтеров MakerBot Replicator, MakerBot Replicator Mini и MakerBot Replicator Z18 (для совместной работы с мобильным приложением на них должна быть установлена версия прошивки 1.4 или новее, причём устройство должно находиться в одной сети Wi-Fi с подключенным к Интернету принтером).

Возможности Makerbot Mobile 1.0 для Android: ➤ приложение обеспечивает поистине революционный уровень удобства и простоты процесса 3D-печати, предоставляя с мобильного устройства

доступ ко всей функциональности принтера Makerbot и подготовку его к работе;

➤ выбор загружаемого цифрового проекта с сервиса Thingiverse или любого файла 3D-модели из «облачной» библиотеки пользователя, использование затем облачных серверов Makerbot для дальнейшей оптимизации, масштабирования и деления на слои своего проекта для 3D-печати;

➤ для печати 3D-модели непосредственно из MakerBot Thingiverse App достаточно найти эту модель и нажать кнопку Print Now. Выбранная модель будет загружена в MakerBot Mobile и сразу отправлена на печать;

➤ контроль и управление 3D-принтером MakerBot Replicator. Встроенная в принтер камера с трехсекундным интервалом отображает изменения в рабочей зоне, позволяя отслеживать про-



➤ Makerbot Mobile 1.0 для Android позволяет управлять 3D-печатью с ладони.

гресс процесса печати, приостановить его или отменить вовсе.

Приложение Makerbot Mobile 1.0 для Android доступно для бесплатной загрузки в Google Play Store. LXF

Сравнение

» Каждый месяц мы сравниваем тонны программ — а вы можете отдыхать!

Векторные редакторы

Ищете лучший векторный редактор под Linux — и для иллюстраций, и для графиков и карт? **Александр Толстой** разглядывает их со всех сторон.



Про наш тест...

Все наши редакторы тестировались на одной и той же 4-ядерной машине с 8 ГБ ОЗУ, на которой работала недавняя версия Elementary OS, основанной на Ubuntu. (Заметьте, для работы с векторной графикой не нужен спецдистрибутив — вы можете с успехом обойтись своим собственным.) Чтобы уравнивать шансы, особое внимание мы уделили 2D-производительности нашей тестовой машины. Для минимизации системных различий мы решили взять чип Intel HD Graphics, принадлежащий к семейству, известному своей стабильностью и достойной, если не выдающейся, производительностью, а также отсутствием проблем с драйверами в Linux. Мы тестировали по программе за раз в стандартных условиях, стараясь брать последнюю стабильную версию каждого приложения, и оценивали удобство в работе, производительность, набор функций и наличие дополнительных материалов в Сети.

Были времена, когда векторные графические редакторы считались специфическими программами, в основном для профессиональных дизайнеров. В наши дни эта планка упала: практически каждый может создавать крутой визуальный контент, что превращает создание векторной графики в увлекательное занятие для взрослых и детей.

Одним из огромных преимуществ векторной графики является ее масштабируемость. В отличие от растровой графики, векторные рисунки не «пикселят» при увеличении: вы можете обрабатывать их как угодно, и ваши рисунки всегда будут

«Большинство приложений поддерживают SVG, промышленный стандартный формат.»

сохранять исходный уровень детализации. При всём своём разнообразии, большинство подобных приложений поддерживают SVG, промышленный стандартный формат для масштабируемой векторной графики.

Для нашего Сравнения мы выбрали пять лучших векторных редакторов, среди которых, наверное, самые известные — *Inkscape* и *LibreOffice Draw* (это

часть популярного офисного пакета). Другие три — *sK1*, разработанный для доредакционной подготовки; *Karbon* из офисного пакета KDE, *Calligra* (ранее называвшегося *KOffice*); и *Xara Xtreme*. Последний является таким чудачеством мира Linux: в 2006 году он был выпущен как программа с открытым кодом, но реализует некоторые функции коммерческого редактора векторной графики *Xara Designer Pro*.

Наша подборка

- » Inkscape
- » Karbon
- » LibreOffice Draw
- » sK1
- » Xara Xtreme

Простота установки

Что нужно, чтобы приложение заработало?

Как и следовало ожидать, простота установки любого приложения зависит от его популярности. Готовый web-браузер предлагается почти в любом дистрибутиве Linux, тогда как научные спецприложения могут иметься только в виде исходного tar-архива.

Векторная графика находится где-то между этими крайностями: то есть если редактор находится в активной разработке, как, например, *Inkscape*, он широко доступен в любом формате пакетов и для любого дистрибутива Linux.

LibreOffice Draw куда меньше обласкан вниманием разработчиков, но поскольку это часть офисного пакета, он обычно имеется почти в каждом дистрибутиве Linux. Для тех, кто предпочитает менеджеру пакетов установку прямо с официального сайта *LibreOffice* (www.libreoffice.org), есть свежие сборки для 32- и 64-битных систем Deb и RPM.

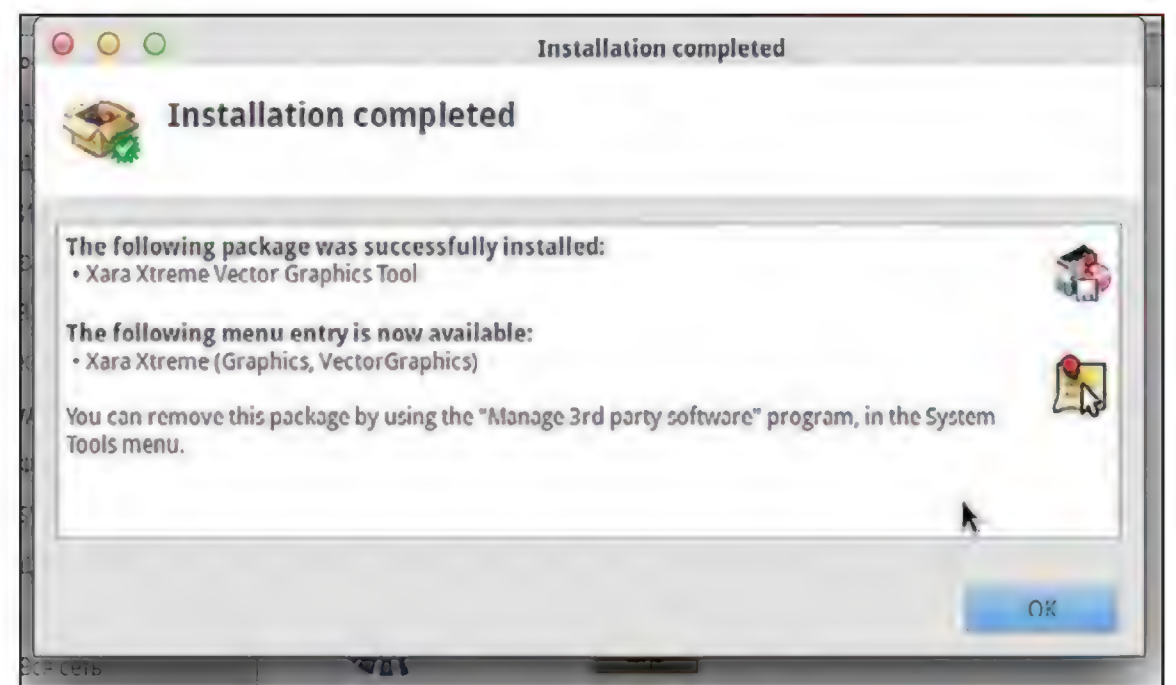
Karbon — тоже часть офисного решения, но *Calligra* используется намного реже *LibreOffice*, и в менее известных

дистрибутивах вы можете не найти готовых пакетов.

sk1 — весьма перспективный проект с открытым кодом, но на момент написания статьи не обновлялся почти два года. Это не столь большая проблема в мире специализированных графических программ с их неспешными циклами разработки, но *sk1* известен только узкой группе разработчиков и членов сообщества: даже не все энтузиасты графики Linux слышали об этом приложении.

Как следствие, *sk1* предлагает множество прекомпилированных пакетов для десятков разновидностей Linux, но, честно говоря, большинство из них устарели. Мы смогли установить пакеты для Ubuntu 11.04 на нашей системе на основе 12.04 LTS, но пришлось вручную справляться с некоторыми зависимостями Python и использовать *dpkg* для установки пакетов *sk1*. Возможно, вам покажется, что лучше скомпилировать его из исходника.

Xara Xtreme даже старше *sk1* — похоже, его сайт последний раз обновлялся



» Windows-подобная программа установки *Xara Xtreme* выглядит мило и аккуратно.

в октябре 2008-го — и вдобавок он поставляется в довольно экзотическом формате Autopackage. В конце 2000-х это был весьма перспективный способ поставки программ Linux, чтобы сохранить их независимость от дистрибутива, но хотя с Autopackage и можно поиграть, велика вероятность осложнений — сейчас, спустя годы забвения, он работает уже не так хорошо.

К счастью, на свете хватает пакетов Deb и RPM. Попробуйте поискать бинарники *xaralx* для своего дистрибутива Linux на <http://pkgs.org>. Если вы выберете свежую версию, приложение заработает без особых проблем.

Вердикт

Inkscape
★★★★★
LibreOffice Draw
★★★★★
Karbon
★★★★★
Xara Xtreme
★★★★★
sk1
★★★★★

» Более активная разработка означает большую доступность.

Удобство в использовании

Быстро ли они позволяют вам начать рисовать?

В данном контексте удобство в использовании означает то, насколько быстро новый пользователь программы сможет освоить предлагаемые функции и начать рисовать, или хотя бы просто изменить параметры приложения.

Inkscape предлагает очень отлаженный и упорядоченный интерфейс с кнопками инструментов слева, панелью цветов

внизу и панелью инструментов справа. Если вам нужно выполнить индивидуальные действия, например, распределить объекты или применить набор фильтров, появляются соответствующие добавочные панели, аккуратно располагаясь с правой стороны окна.

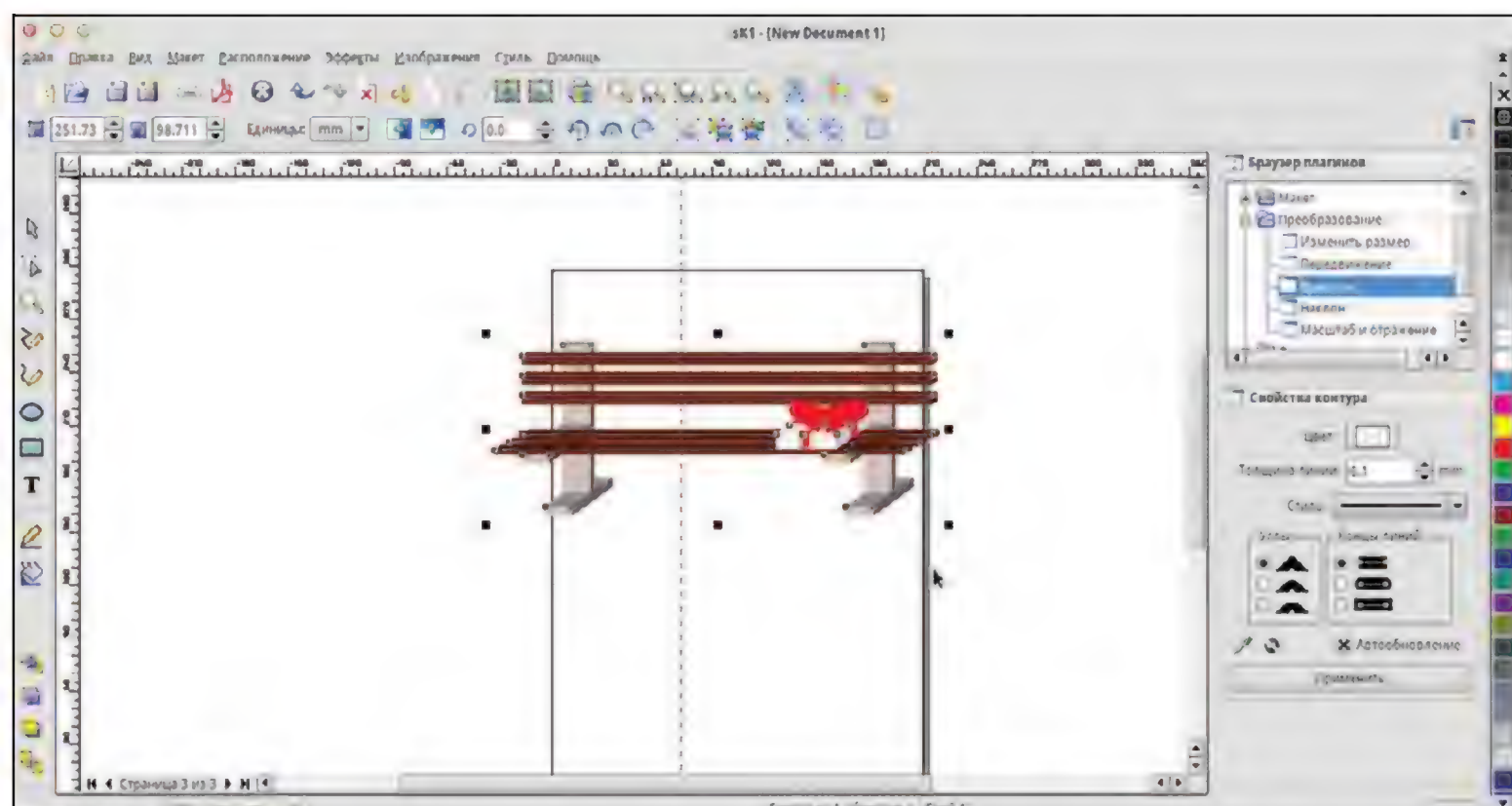
В *LibreOffice Draw* основная панель инструментов располагается внизу окна,

и можно добавлять инструменты через дополнительные панели меню View. Новые наборы инструментов будут появляться в виде плавающих панелей, которые вы можете разместить, где хотите. Поскольку это офисное приложение, *Draw* предлагает огромную панель Pages на левой стороне окна.

У *Karbon* незахламленный интерфейс с традиционной структурой (инструменты слева, опции справа), однако многие функции трудно найти.

Xara Xtreme весьма дружелюбен, но выглядит не слишком аккуратно из-за своего устаревшего GUI, основанного на wxWidgets, а в опциях редактора не хватает многих значков.

sk1 — очень стабильное приложение, но ему не хватает многих функций, или в этих функциях вместо настоящих разделов отображаются заполнители, особенно в настройках программы. Индивидуальные горячие клавиши для увеличения масштаба или работы с объектами также потребуют некоторого времени на изучение.



» Несмотря на состояние незавершенности, *sk1* вполне можно пользоваться.

Вердикт

Inkscape
★★★★★
LibreOffice Draw
★★★★★
sk1
★★★★★
Xara Xtreme
★★★★★
Karbon
★★★★★

» Некоторые приложения кажутся менее законченными и стабильными, чем другие.

Имеющиеся инструменты

Сколько предлагается инструментов и полезных опций?

Основной критерий — разнообразие инструментов, имеющихся в каждом редакторе: инструментов рисования, возможностей редактирования узлов и опций для работы с изображениями, внешними объектами и страницами, плюс уникальные и интересные инструменты для привлечения внимания

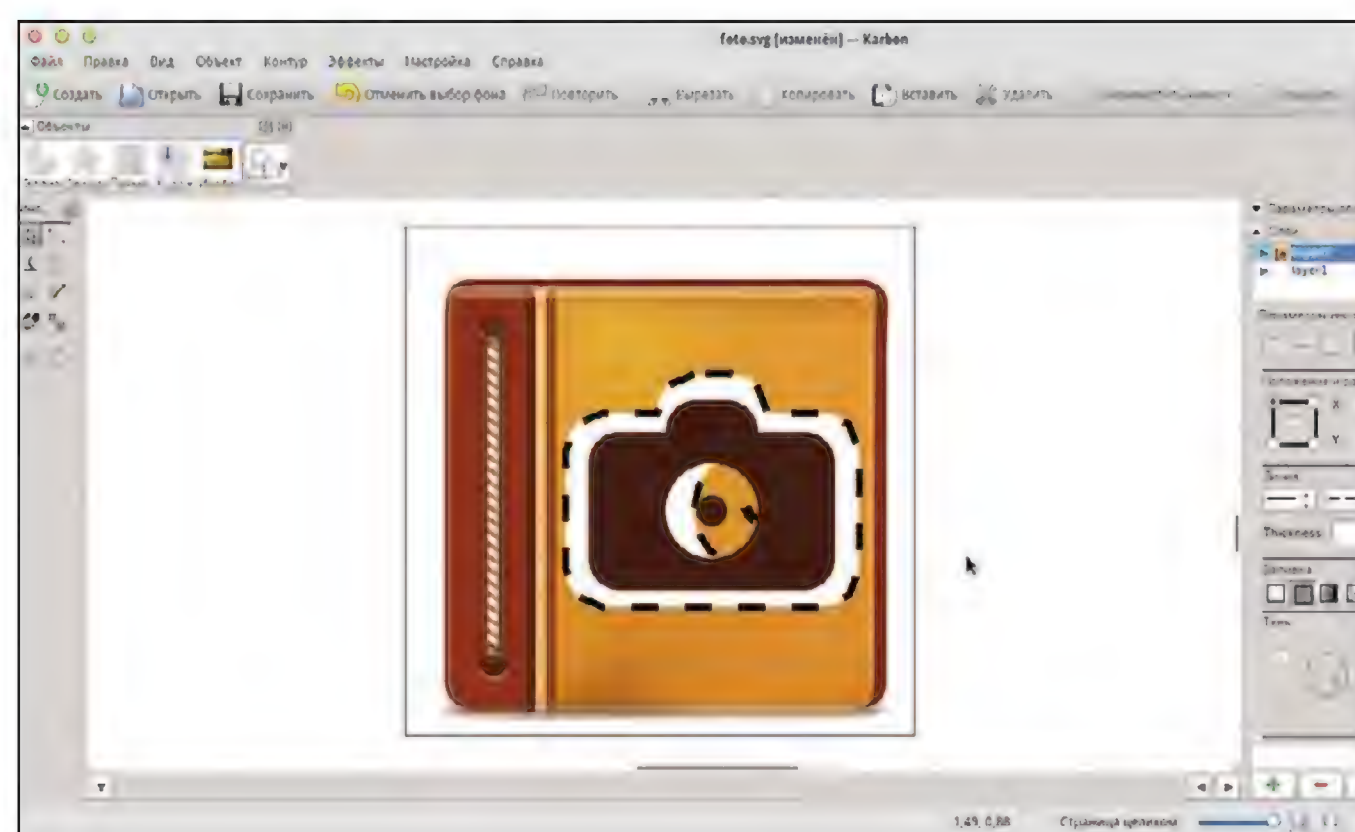
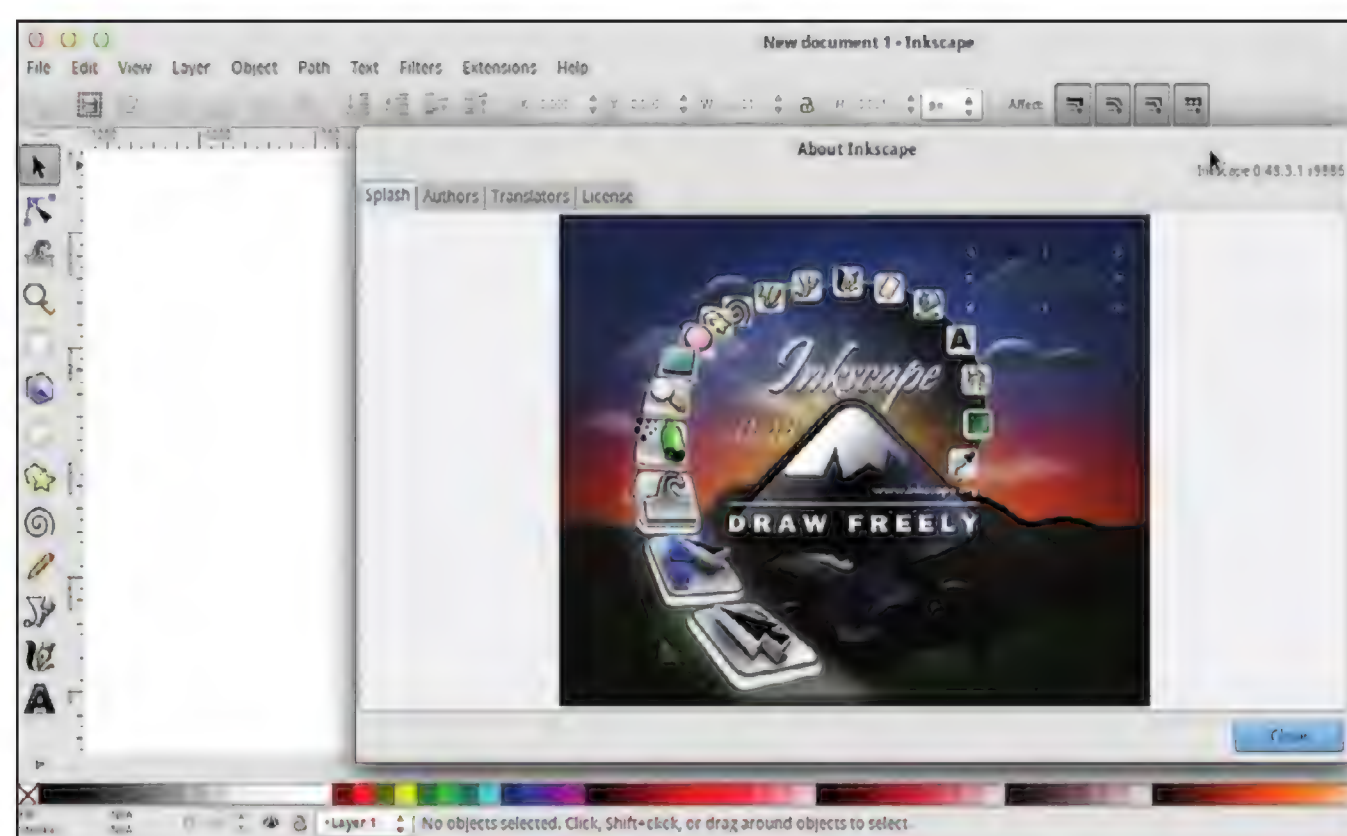
поклонников векторной графики. У каждого из наших участников свой специфический уклон, который мы отметим по ходу рассмотрения; но, с другой стороны, понятно, что чем шире набор инструментов — тем лучше.

Конечно, богатство приложения функциями, будь оно специализированным или же общего

назначения, в основном определяется активностью его разработки: в конце концов, добавление каждой кнопки на панели инструментов или пункта меню требует времени, и более сильное сообщество или оплачиваемая команда программистов обычно способны создать более впечатляющий продукт.

Inkscape ★★★★★

Inkscape — самый богатый функциями и поэтому самый гибкий редактор векторной графики для Linux. По части объединения и разъединения узлов, работы с путями объектов, наложения, слияния и извлечения объектов он сопоставим с проприетарным *Adobe Illustrator*. Кроме того, он предлагает массу опций рисования. Здесь имеется инструмент свободного рисования (карандаш), настраиваемые кисти и множество готовых фигур. *Inkscape* умеет также очерчивать растровую графику и конвертировать ее в векторные объекты. И он отлично работает с текстом: поддерживается сложное форматирование символов и абзацев, отступы, кернинг и даже проверка правописания. Текст вы можете привязать к кривой, или сделать индивидуальное заполнение и обводку, или применить любое сочетание художественных эффектов — и при этом он останется редактируемым. Кроме того, можно в любой момент превратить текст в контур.



Karbon ★★★★★

Karbon — векторный напарник *Krita*. Оба они являются частью *Calligra Suite*, и оба изначально предназначены для художников. Но в отличие от *Krita*, возможности *Karbon* весьма скромны. Мы сочли, что это чисто базовый редактор векторной графики с несколькими расширенными опциями. В нем есть ряд простых инструментов фигур, но они размещены в панели объектов справа, а не на основной панели инструментов. У *Karbon* вполне достойный набор инструментов для работы с узлами, линиями и кривыми, и он умеет привязывать и распределять объекты и даже создавать слои. Есть в нем также инструмент «художественный текст», хотя на самом деле он оказывается не особо художественным: это обычный ввод. Вы можете выравнивать текст, кое-что выделить и привязать текст к кривой, но сверх этого поддерживается мало опций форматирования. Обводка растровых изображений для превращения их в векторные — еще одна стандартная функция, которой в *Karbon* нет.

Фильтры и эффекты

Что даст вам каждое приложение помимо базовых функций рисования?

Создав некую векторную графику, вы наверняка захотите добавить в нее эффекты — хоть для web-графики, хоть для проектов оффлайн. Коллекция эффектов *Inkscape* намного превосходит остальные — здесь десятки хитростей относительно форм и объемов, применимых к редактируемой кривой или тексту, потрясающая морфология, материалы и текстурные фильтры для векторных объектов, а также ряд художественных фильтров для растровых

изображений. Эффекты помещены в меню *Filters*, и есть еще специальная панель редактирования *Filter Effects*, где можно объединять эффекты и сортировать их. *LibreOffice Draw* тоже предлагает широкий набор эффектов, но они скорее канцелярские, чем художественные. Неплохи эффекты шрифтов (клонированные с *WordArt* из *Microsoft Office*), возможность привязать текст к окружности, инструменты экструзии и добавления тени. Они, однако, мало подходят для творчества.

В *Karbon* есть меню *Effects*, но там только 4 пункта для базовых модификаций контура, и нельзя работать с SVG-эффектами при открытии файла, созданного в другом приложении. *sK1* далеко не ушел: добавлять нечего, и нет поддержки импорта SVG-эффектов. *Xara Xtreme* честно предупреждает о ранней стадии своей разработки и не поддерживает импорт файлов SVG вообще, но имеет интересные инструменты: эффект выпуклости, лекало, тень и прозрачность... это лучше, чем ничего.

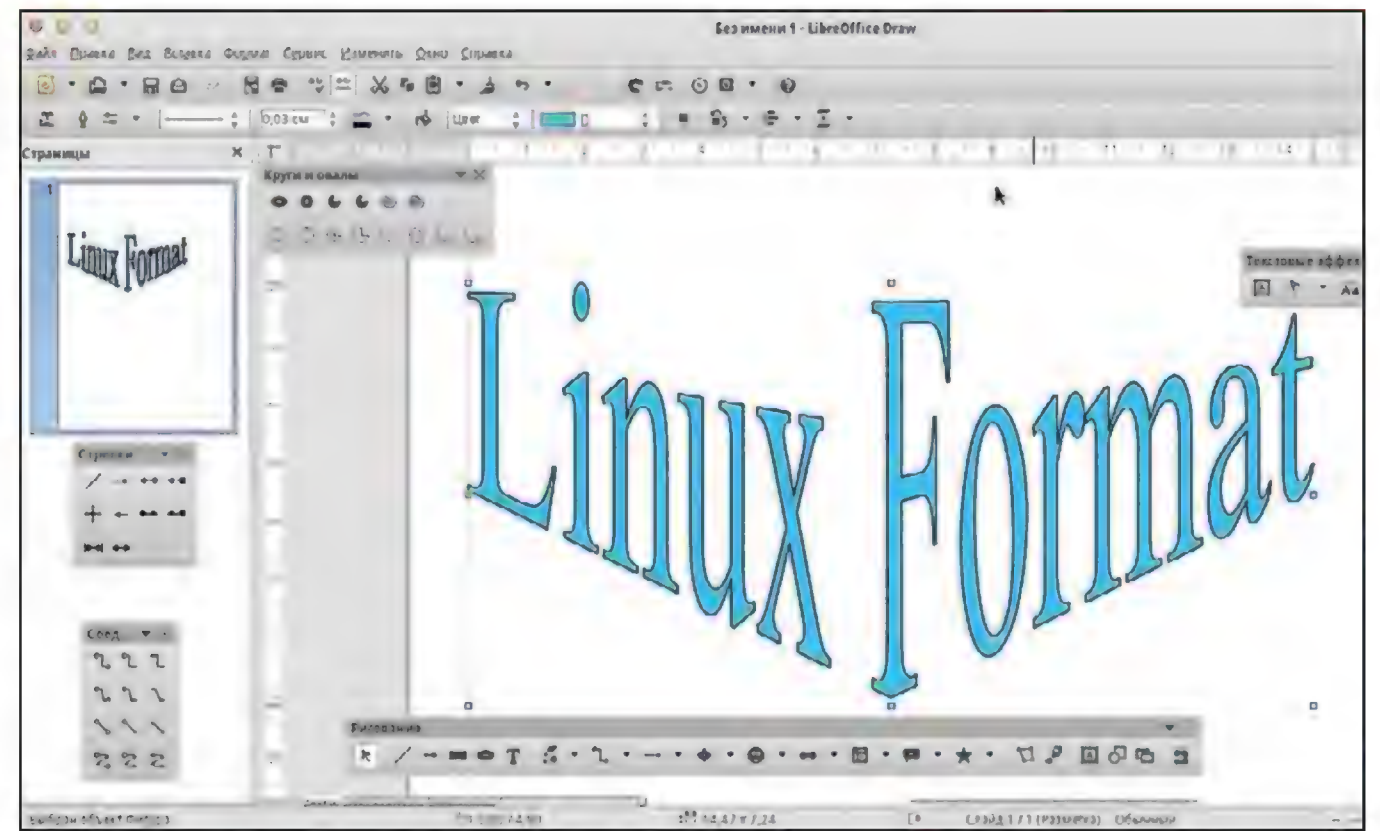
Вердикт

Inkscape ★★★★★
LibreOffice Draw ★★★★★
Xara Xtreme ★★★★★
Karbon ★★★★★
sK1 ★★★★★

» Отличные эффекты *Inkscape* выделяют его в отдельный класс.

LibreOffice Draw ★★★★★

В *Draw* наблюдается заметный сдвиг в сторону деловой графики: он очень подходит для разработки схем, планов, графиков, и обладает весьма солидным модулем работы с растровой графикой. Есть инструменты для свободного рисования фигур, позволяющие создавать многоугольные объекты (например, для карт), но вы мало что можете сделать с узлами и пересечением объектов. Он явно предназначен для создания простых фигур и упорядочивания отношений между ними, поэтому вы можете перетаскивать объекты, рисовать красивые связующие стрелки и размещать объекты, но будет практически невозможно — или, по крайней мере, очень сложно — нарисовать реалистичный образ. *Draw* очень хорош для простой графики и довольно неплох для зарисовок и чертежей, что делает его превосходной программой для бизнеса, но даже это можно было бы расширить большим количеством художественных средств, таких как кисти и внешние эффекты.

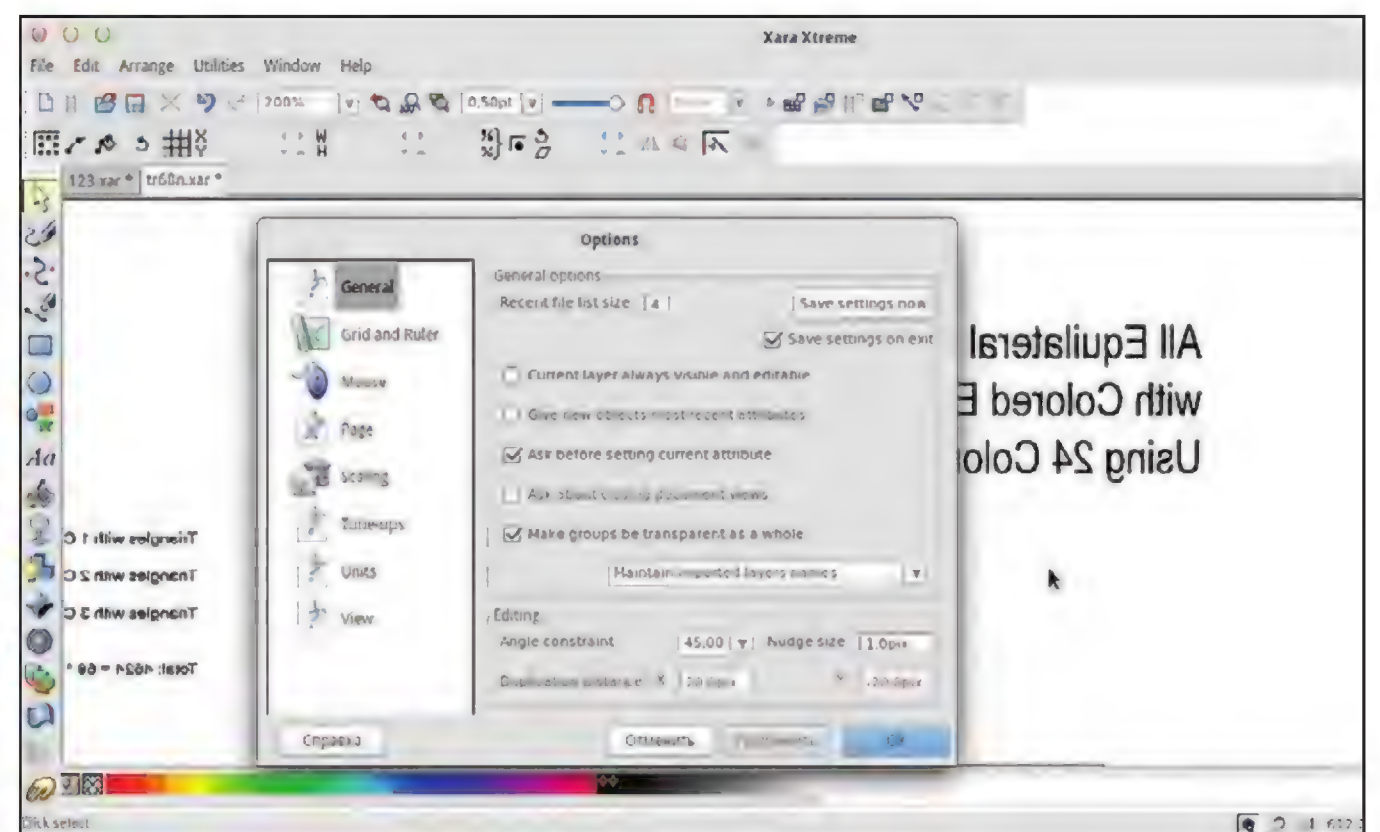


sk1 ★★★★★

SK1 весьма успешно воссоздает ряд основных функций профессиональной программы DTP [Desk Top Publishing — настольная издательская система] в проекте открытого кода. Этим функций множество; к ним относятся поддержка цвета CMYK, разделения, управление цветом ICC и вывод результата в PDF. Левая часть экрана содержит браузер плагинов с очень симпатичной древовидной структурой, благодаря чему *sk1* выглядит как гибрид известного приложения DTP *Scribus* и *Skencil*, его предшественника. Здесь предусмотрены инструменты для рисования, но они довольно-таки примитивны, как и опции для работы с узлами и кривыми. *SK1* больше подходит для работы с объектами и подготовки графики для печати, а не для художественного или творческого использования. Кроме того, он оставляет ощущение незавершенности, поскольку многих функций в нем не хватает, а это не здорово.

Xara Xtreme ★★★★★

Коммерческая версия *Xara Xtreme* заявляет, что функций в ней не меньше, чем в *Inkscape*; и, похоже, их даже больше. Но *Xara Xtreme* для Linux — лишь бледная тень коммерческого оригинала. Правда, в нем немало функций и инструментов для работы с узлами, кривыми, фигурами и перетеканием, форматирование текста, редактирование цвета и градиентной заливки, и многое еще. В общем, он производит впечатление зрелого, позволяет создавать сложные произведения из множества тщательно созданных векторных объектов, предлагает хороший набор векторного клип-арта и имеет встроенный поиск растровой графики. Многие функции не реализованы (скажем, продвинутые инструменты свободного рисования и кистей или растровый редактор), и все же *Xara Xtreme* — неплохой выбор для цифровых художников. Здесь отличные эффекты выпуклости, настраиваемые тени и точное управление линиями и кривыми, плюс поддержка кернинга и разметки для текста.



Совместимость форматов

Хороша ли поддержка промышленных стандартов векторных форматов?

В публикациях и векторной графике есть ряд основных файловых форматов, существенных для надежного хранения данных и взаимобмена файлами: Portable Document Format (PDF) и Encapsulated PostScript (EPS) для сохранения предпечатных страниц; Scalable Vector Graphics (SVG) для хранения кривых и контуров; и проприетарные форматы AI и CDR для работы с файлами Adobe Illustrator и CorelDRAW. Форматы PDF и SVG считаются обязательными. Кроме *sk1*,

в каждом редакторе есть поддержка импорта PDF, но *Karbon* не справился с парсингом нашего тестового файла PDF, а *Xara Xtreme* импортировал его как растровое изображение (все слилось и растеризовалось). А вот у *LibreOffice Draw* очень достойная программа импорта и экспорта PDF. Идеальную поддержку SVG обеспечивает *Inkscape*; неплохую — *LibreOffice Draw*, *Karbon* и *sk1* (хотя эти трое поддерживают только базовые функции SVG). В *Xara Xtreme* нет поддержки SVG и вообще слаба

поддержка форматов. В свое время проект *sk1* создал *UniConverter*, инструмент, способный считывать массу проприетарных форматов, в том числе AI до v9, EPS, CorelDRAW X3 и X4, и многие другие (см. <http://bit.ly/UniConverter>) — это лучшее решение с открытым кодом для данной задачи. В *sk1* оно применяется по умолчанию, а в *Inkscape* определяется автоматически. У *Xara Xtreme* своя реализация поддержки AI и EPS, и можно спокойно писать в этих форматах; но чтение не столь надежное.

Вердикт

Inkscape
★★★★★
LibreOffice Draw
★★★
sk1
★★★
Karbon
★
Xara Xtreme
★

» *SK1* мог бы набрать больше баллов; но уж очень плохо поддержка PDF.

Производительность

Как наши приложения справляются с тяжелым грузом векторных файлов?

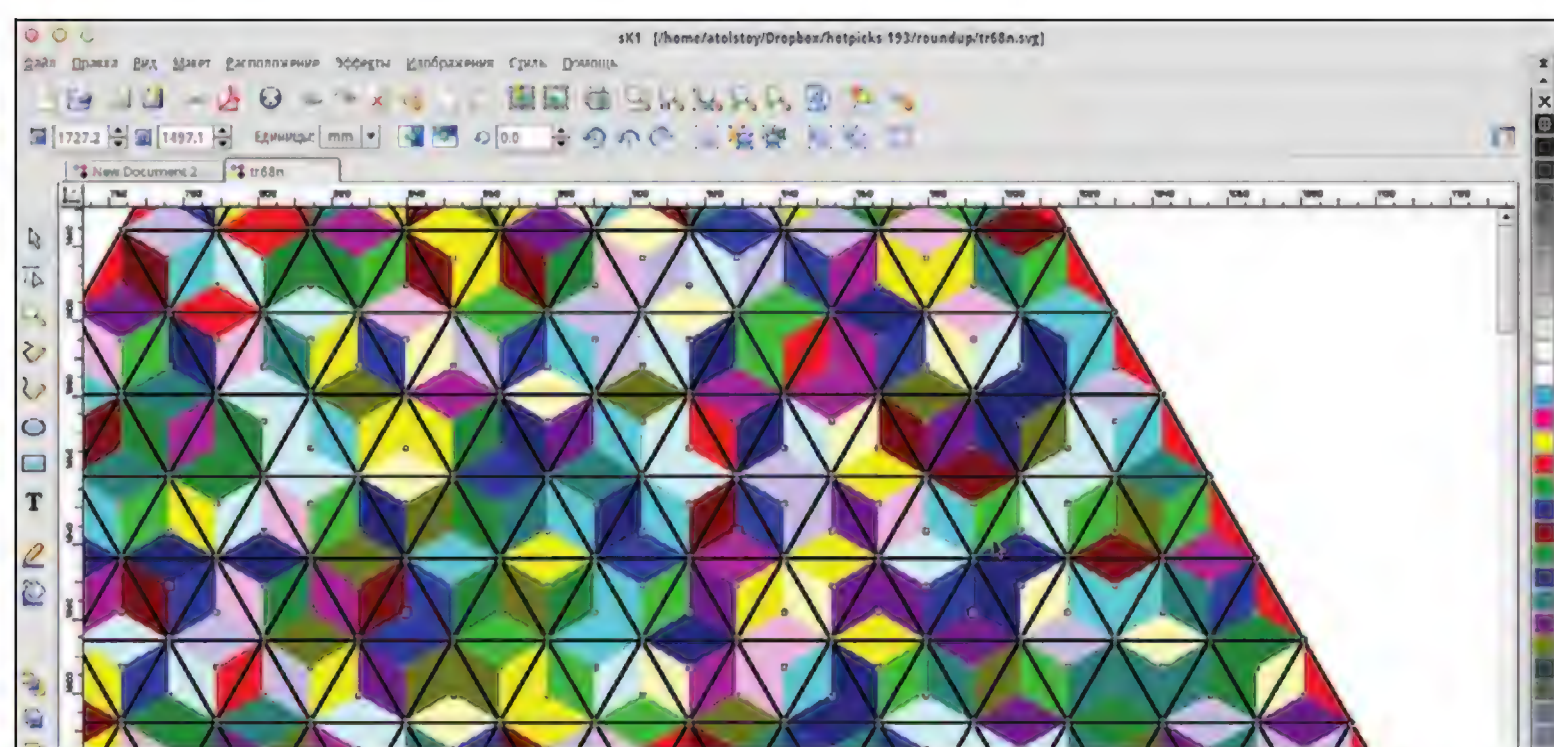
В отличие от растровой графики, даже в очень сложной векторной графике файлы небольшие, и работа с ними является тестом производительности процессора, а не скорости чтения вашего жесткого диска. Наш образец, файл SVG, несмотря на размер всего 2 МБ, содержал 18496 объектов, большая часть которых граничила друг с другом. Во время теста надо было открыть файл и затем

зеркально отобразить все его объекты. Для некоторых наших редакторов обработка файла оказалась сложной задачей, но результаты удивили. *Inkscape* не слишком хорошо проявил себя: он медленно выполнял увеличение сложных контуров SVG и в целом отличался неудобством в работе. Тестовый файл он открыл лишь спустя 40 секунд, и любые действия приводили к значительным задержкам.

И *LibreOffice Draw*, и *Karbon* крайне медленно открывали тестовый файл или выполняли любые модификации объектов. На выполнение нескольких простых действий у нас ушло несколько минут, и мы бы не хотели использовать эти приложения для более тяжелой векторной графики.

С *sk1* — совсем другая история. Хотя *sk1* использует тот же движок рендеринга Cairo, что и *Inkscape*, его производительность и время отклика были во много раз лучше, и у нас получилось весьма комфортно поработать с нашим тестовым файлом.

Xara Xtreme не умеет импортировать SVG, поэтому мы конвертировали тестовый файл в PDF в *Inkscape*. *Xara Xtreme* использует собственный движок рендеринга, известный своей быстротой и хорошей оптимизацией. Так и произошло с нашими бесконечными маленькими треугольниками. Выделение и зеркальное отображение более 18000 объектов заняло всего несколько секунд — умопомрачительная производительность.



» Эта броская композиция из более 18000 векторных треугольников повергла на колени большинство наших редакторов. Но некоторые данный тест прошли.

Вердикт

sk1
★★★★★
Xara Xtreme
★★★★★
Inkscape
★★★★
Karbon
★
LibreOffice Draw
★

» Xara Xtreme и sk1 показали здесь удивительную скорость.

Документация и поддержка

Много ли полезной информации вы получите или найдете онлайн?

Поиск ответов на свои вопросы чрезвычайно важен, и частенько именно он является причиной выбора той или иной программы. Посмотрим же, насколько хороша документация наших векторных редакторов.

Inkscape имеет отличный онлайн-справочник, wiki на <http://wiki.inkscape.org> и набор официальных и сторонних руководств. Даже если вы никогда не работали с *Inkscape*, вы полюбите великолепную графику его примеров и научитесь шаг за шагом воссоздавать их. Поиск ответов на вопросы в Сети выведет вас на большое количество форумов и тем, посвященных этому первоклассному редактору.

LibreOffice Draw предлагает симпатичный, но скромный раздел основного портала *LibreOffice* на <http://help.libreoffice.org>, с руководствами и справочниками по основным инструментам рисования и по работе с объектами; некоторые из них размещены на порталах сторонних справочников. Объем документации соответствует набору функций программы.

В *Karbon* есть только маленькая страница внутри общей документации по *KOffice* (бывший *Calligra Suite*). В *Karbon* есть руководство по работе с художественными формами текста, и это всё. Поиски в Сети хоть чего-то по теме привели нас к неуместным сайтам про бензин и отопление.

sk1 последние годы не выказывал особой активности, но есть весьма неплохой форум поддержки для *sk1* и *UniConverter* (<http://sk1project.org/forum>), с ответами на наиболее общие вопросы. Быстрый поиск также выявил достаточное количество обзоров от сообщества и вводных постов по поводу установки и функций, которые могут очень пригодиться. Учитывая скромный набор функций *sk1*, мы бы сказали, что ни одна крупная проблема не осталась без внимания.

Поиски материалов по *Xara Xtreme* выявили множество всяких ссылок,

относящихся к коммерческим инструментам *Xara*. Но заглянув на сайт <http://xaraxtreme.org>, вы обнаружите тонны тщательно отобранной документации. Там имеются очень подробные и полезные FAQ и ссылки на архивированную страницу форума программы (<http://bit.ly/Xara-Forum>). Хотя сам редактор остался в незавершенном состоянии, его документация весьма зрелая и выглядит профессионально. И, опять же, соответствует набору функций.



» Одиноким справочник *Karbon* не может конкурировать с всеобъемлющей библиотекой, доступной для *Inkscape*.

Вердикт

Inkscape
★★★★★
Xara Xtreme
★★★★★
LibreOffice Draw
★★★★
sk1
★★★★
Karbon
★★

» Объем поддержки соответствует набору функций каждого приложения.

Редакторы векторной графики

Вердикт

Последнее Сравнение редакторов векторной графики было в LXF79, но и на этот раз у нас тот же победитель, что и в 2006 г. *Inkscape* работает весьма зрелищно, справляясь с большинством аспектов векторной графики. Он предлагает незахламленный и интуитивный интерфейс, отличный набор инструментов для рисования кривых и почти бесконечное число привлекательных фильтров и эффектов. *Inkscape* развивался долгие годы, превратившись в «почти профессиональное» приложение, пригодное и для художественных целей, и для основных задач DTP. Благодаря *UniConverter*, он может импортировать многие (если не все) коммерческие разновидности форматов AI и CDR, а если у вас вдруг возникнут проблемы, вы легко найдете поддержку, а также многочисленные обсуждения решений онлайн.

Xara Xtreme не отличается подобной полнотой, но заслуживает второго места

благодаря своему быстрому движку рендеринга, достойной поддержке форматов (особенно для экспорта) и хорошей документации. Трудно сказать, возьмется ли кто-нибудь за разработку программы, но *Xara Xtreme* остается удобной в использовании, быстрой и свободной.

LibreOffice Draw неплохо справляется со своей работой; но он не предназначен для художественных задач и предлагает относительно скромный набор инструментов. Иными словами, он будет очень хорош для создания графиков, схем и простых чертежей. Он отличается стабильностью и солидностью, и его преимущество — принадлежность к пакету *LibreOffice*, что отчасти компенсирует его ограничения.

На четвертом месте — *sK1*, решение с сильным уклоном в DTP, с удобными инструментами векторной графики, быстрым и отзывчивым движком рендеринга и фантастической



утилитой *UniConverter*. Делается попытка разработки преемника *sK1* под названием *PrintDesign*, однако работа ведется медленно — последнее демо вышло в октябре 2013 и было для Ubuntu 13.04. *sK1* явно заслуживает более пристального внимания новых пользователей и разработчиков.

И, наконец, хоть и предназначен он для художников, *Karbon*, похоже, подотстал от времени. В 2006 г. в нем было больше функций, чем сейчас, и в данный момент у него нет никаких шансов побить *Adobe Illustrator* в мире Linux.

» В Сети можно найти справочники по созданию подобных впечатляющих произведений (в данном случае, в *Inkscape*).

«Inkscape подходит как для художественных работ, так и для базового DTP.»

I**Inkscape** ★★★★★

Сайт: <http://inkscape.org> Лицензия: GNU GPL Версия: 0.48.5

» Приятное в работе отличное приложение для редактирования векторной графики со множеством функций.

IV**sK1** ★★★★★

Сайт: <http://sk1project.org> Лицензия: GNU LGPL Версия: 0.9.1

» Быстрый и довольно простой в работе, но не очень богатый функциями.

II**Xara Xtreme** ★★★★★

Сайт: www.xaraxtreme.org Лицензия: GNU GPL Версия: 0.7

» Незаконченный, но многообещающий профессиональный редактор векторной графики.

V**Karbon** ★★★★★

Сайт: www.calligra.org/karbon Лицензия: GNU LGPL Версия: 2.8.6

» Проклятие KOffice по-прежнему тяготеет над этим небольшим приложением.

III**LibreOffice Draw** ★★★★★

Сайт: www.libreoffice.org Лицензия: GNU LGPL/MPL Версия: 4.3.4

» Солидный и крепко сбитый редактор для любых офисных потребностей, связанных с векторной графикой.

Обратная связь

Вы используете отличную программу векторной графики, о которой хотите рассказать всему миру? Начните с нас: lxflatters@futurenet.com.

Рассмотрите также...

Для нашего Сравнения мы выбрали пять самых выдающихся редакторов векторной графики для Linux. Но есть и другие достойные приложения. Стоит обратить внимание на *Sketsa SVG Editor* (\$89, www.kiyut.com), вдохновленный *SCADA* редактор *GLIPS Graffiti* (<http://glipssvgeditor.sourceforge.net>) или *Xfig* (www.xfig.org) — старомодный, но по-прежнему активный

и удобный. Даже давно забытые программы типа *Sodipodi* порой заслуживают внимания, в особенности для образовательных целей.

Вы, вероятно, заметили, что большинство редакторов здесь кросс-платформенные, но можно пойти еще дальше и выбрать векторный онлайн-редактор. Взгляните на *Janvas* (www.janvas.com): расширение *Chrome/Chromium*, в первую очередь

предназначенное для создания web-графики. Немало и других, в том числе *SVG-edit* от Google и *Vector Magic* с открытым кодом (www.vectormagic.com). Большинство этих редакторов просты в работе и освоении и предлагают помощь новичкам. Выбор велик, и включает очень хорошие программы с открытым кодом на любой вкус — можете принять участие. **LXF**

СТРОИМ В LINUX МЕДИА-СЕРВЕР

Превратите свой ПК в хаб для воспроизведения всех своих фильмов, ТВ-программ, музыки и фотографий потоком на любых устройствах. Ник Пирс вручает вам пульт.



Для создания медиа-сервера есть веская причина: доступность всех мультимедиа-файлов, которые есть в наличии. Медиа-сервер позволяет не перекапывать на полке DVD и CD и не быть прикованным к одной точке в доме, если захочется посмотреть фильм, поделиться фотографией или послушать музыку.

Все, что вам нужно — кроме, разумеется, самих мультимедиа-файлов — подходящий компьютер с Linux, который станет сервером. Хотите ли вы довериться новенькому компьютеру или дать жизнь старому, но вполне исправному настольному ПК или ноутбуку, наша статья вам подойдет. Ваша главная проблема — выбор медиа-сервера. В нашей статье мы сосредоточимся на *Kodi*, ранее

известном как *XBMС*, и расскажем о плагинах, заполняющих некоторые его пустоты, а также укажем альтернативы, если вы захотите пожертвовать принципом свободного ПО ради больших возможностей и встроенных функций. Затем мы расскажем вам все, что нужно

«Мы расскажем вам все, что нужно знать о настройке библиотек медиа.»

знать о настройке библиотек медиа, от именования и структурирования файлов до добавления их на ваш сервер. Мы рассмотрим, как интегрировать в медиа-сервер ваши библиотеки фотографий, даже несмотря на то, что *Kodi* (пока) этого не поддерживает, и как организовать

поток воспроизведения музыки через динамики AirPlay и другие беспроводные устройства. По окончании настройки медиа-сервера останется только разобраться, как обращаться к файлам медиа с других устройств. В последней части нашей статьи мы пройдемся по различным вариантам, рассказав о приложениях и приемах, которые позволяют смотреть видео и слушать музыку на самых разнообразных устройствах, от другого компьютера или мобильного устройства (с Android или iOS) до Smart TV, телевизионной приставки, выделенного медиа-центра на Raspberry Pi или устройства потокового воспроизведения, включая Google Chromecast и Apple TV. Итак, вы готовы менять формат использования медиа у себя дома? Тогда читайте дальше.

Выбираем сервер

Выбираем подходящее «железо» и ПО для медиа-сервера.



Первый вопрос, который нужно себе задать — как именно будет организован медиа-сервер. С точки зрения железа ответ на него сводится к двум основным вариантам: воспользоваться наличным компьютером или ноутбуком или выбрать специализированное устройство, с единственным назначением: предоставлять мультимедиа-данные везде и всем, кто ни попросит. Преимущества использования имеющегося компьютера очевидны: экономия и минимум настроек. Просто установите медиа-сервер на свой компьютер и сразу переходите к стр. 34.

Однако затем спросите себя, сильно ли повлияет медиа-сервер на повседневные задачи. Если вы хотите, чтобы другими задачами он не занимался, нужен выделенный компьютер. Если у вас под рукой есть старый ноутбук или компьютер, вы еще и сэкономите. Просто убедитесь, что у него хватит прыти для решаемых задач.

Вам нужен двоядерный процессор частотой 2 ГГц с видеокартой, поддерживающей аппаратное декодирование видео (Nvidia GeForce серии 8 или выше; GMA X4500HD или выше от Intel или Radeon R700 (HD 4000) или выше от AMD), а это почти любая видеокарта с интерфейсом PCI-e. Что касается оперативной памяти, 4 ГБ более чем достаточно.

Также подумайте о замене загрузочного диска на дешевый твердотельный, быструю SD-карту или USB-диск (32 ГБ достаточно; подойдет даже 4 ГБ, если вы выберете OpenELEC), при этом сами файлы мультимедиа должны храниться отдельно, либо на втором внутреннем диске, либо на USB-или даже NAS-диске.

Если вы хотите купить новый компьютер, то дешевый вариант — Raspberry Pi, но это скорее медиа-проигрыватель, чем сервер. Вместо него взгляните на неттопы от Zotac или Acer или соберите собственный мини-компьютер с материнской платой небольшого форм-фактора Mini-ITX и подходящим корпусом, таким как Antec ISK 110 с креплением VESA.

Подходящее ПО

Если ваш сервер несет двойную нагрузку в качестве медиа-сервера под главным телевизором или под телевизором в спальне, то для удобства стоит воспользоваться беспроводной клавиатурой и мышью. Также подумайте о HomePlug, если сервер не подключен напрямую, особенно если вы планируете поток HD-видео (для воспроизведения без торможения выберите модель с 500 или 600 Мбит/с).

При выборе ПО для структурирования музыки, фильмов и фотографий и потокового воспроизведения на все остальные устройства вариантов масса. Мы взяли *Kodi*, более известный по своей предыдущей инкарнации — *XBMBC*. Даже несмотря на то, что это скорее медиа-центр, он может служить и медиа-сервером, и его главное преимущество — в том, что он бесплатен, имеет открытый исходный код и не содержит ничего лишнего. *Kodi* можно установить в существующую систему Linux, но если вы решили создать выделенный медиа-сервер, выберите одну из версий, встроенных в отдельные ОС.

Если вам нужен доступ к Linux в этой отдельной ОС (на основе Ubuntu LTS), выберите XBMC-buntu (<http://kodi.wiki/view/XBMCbuntu>), а если у вас менее мощный компьютер и нужен только *Kodi* без мороки с настройками, выберите OpenELEC (<http://openelec.tv>).

Независимо от того, будете ли вы устанавливать *Kodi* на существующий компьютер с Linux или на выделенный сервер, установка проста; ссылки на соответствующие репозитории и исходный код, а также инструкции по установке можно найти в вики *Kodi* (<http://bit.ly/XBMC4Linux>). После установки все готово к запуску сервера.

Альтернативные медиа-серверы

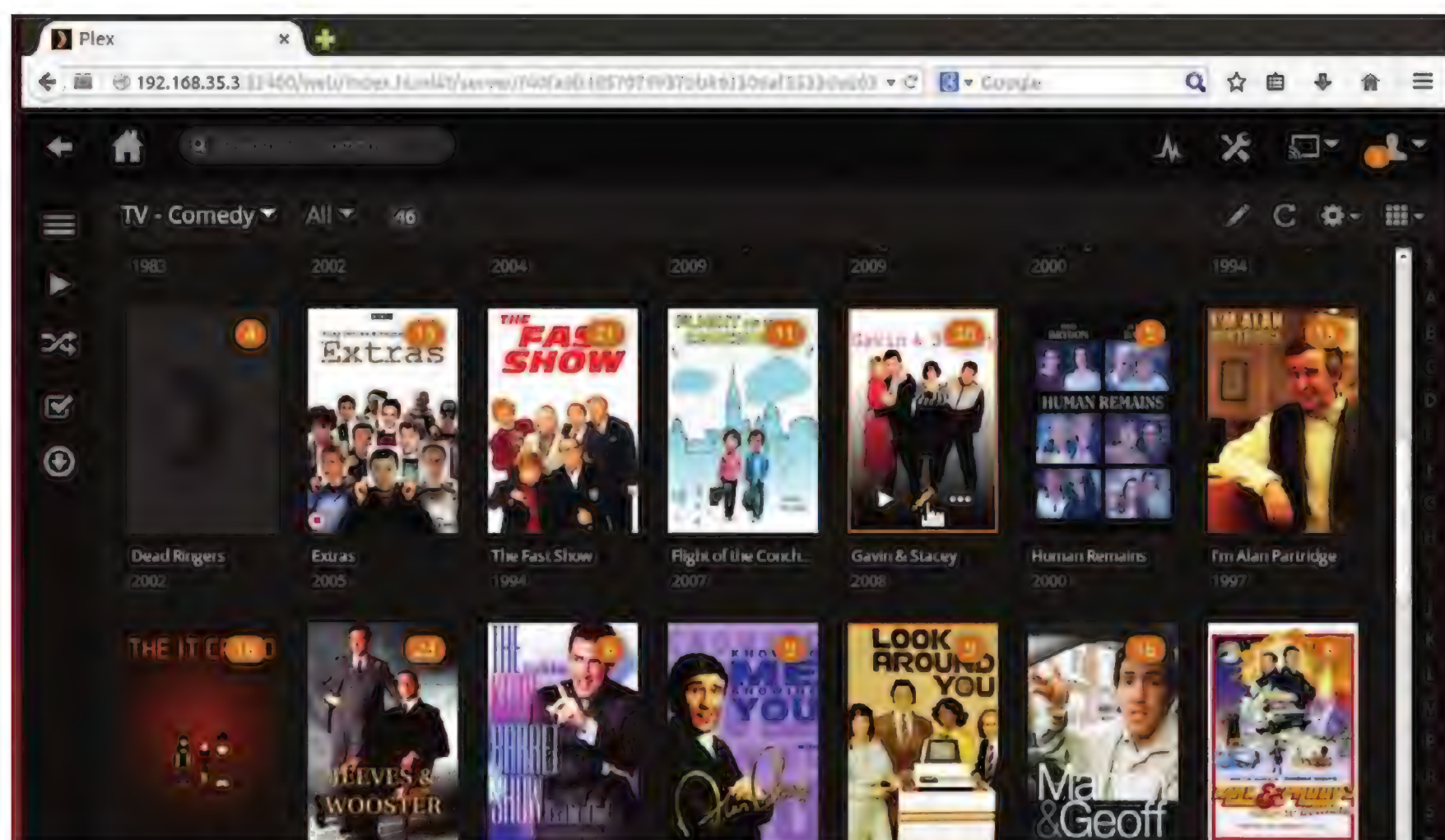
Хотя в нашей статье мы рассказываем только о *Kodi*, существует масса других медиа-серверов, и некоторые из них могут лучше подойти для решения ваших задач. Самая заметная альтернатива — *Plex Media Server* (<https://plex.tv>); если честно, это более мощный и гибкий продукт и может показаться привлекательнее тому, кто предпочитает достаточно простую процедуру установки и не хочет тратить время на обход подводных камней из-за некоторых отсутствующих в *Kodi* возможностей. Однако его исходный код закрыт, и все приложения с максимумом возможностей (включая мобильные) платны. Но он повсеместно поддерживается, и можно установить на свой сервер *Plex Media Server* вместе с *Kodi* и связать их друг с другом через бесплатное дополнение *PlexXBMC*.

На корону *Plex* также покушается *MediaBrowser* (<http://mediabrowser.tv>), и теперь он доступен как для Ubuntu, так и для Debian (в бета-версии). Он похож на *Plex* тем, что имеет приятный интерфейс для широкого набора медиа-форматов и поддерживает несколько типов домашних и мобильных устройств; но часть приложений опять же платные.

Если вы пытаетесь установить медиа-сервер прямо на NAS-диск в Linux, стоит подумать о *MediaTomb* (<http://mediatomb.cc>) или *Serviio*

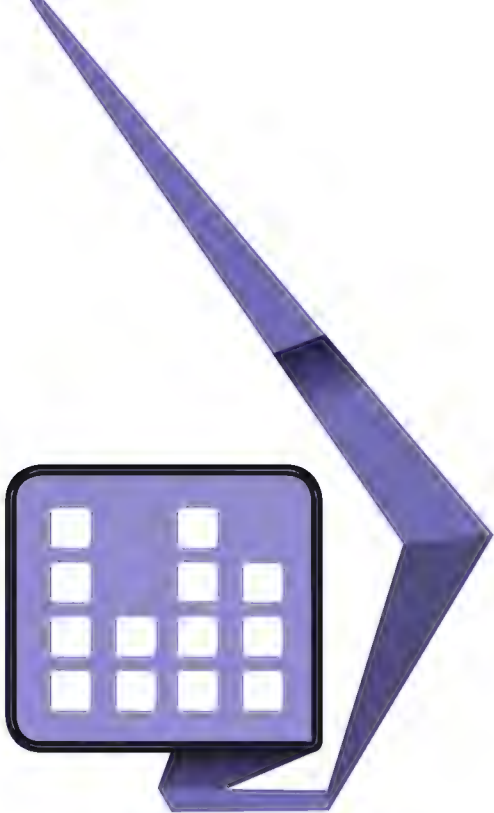
(<http://serviio.org>). Оба поддерживают все совместимые устройства Universal Play and Play (UPNP), но для доступа к устройствам, не относящимся к DLNA (Digital Living Network Alliance), в *Serviio* придется купить лицензию Pro.

Если вас главным образом интересует потоковое воспроизведение, рассмотрите *Subsonic* (<http://subsonic.org>). Это web-сервер, позволяющий обращаться к ряду поддерживаемых устройств как в сети, так и за ее пределами.



➤ *Plex Media Server* — известная альтернатива *Kodi*, но за продвинутые функции взимает плату.

Установка и настройка



Самое сложное — организация и настройка библиотек медиа, и вот что надо делать.

После установки *Kodi* — отдельно или в составе OpenELEC или XBMCbuntu — первым делом структурируйте свои файлы медиа. Их можно хранить на любом диске, доступном серверу, но мы рекомендуем по возможности делать это на локальных дисках: потоковое воспроизведение файлов с NAS-диска может привести к проблемам, особенно для HD-видео.

Где бы вы ни хранили свои файлы, важно правильно структурировать их, т.е. создать логичную структуру каталогов, где файлы хранятся, и правильно назвать файлы (см. ниже). Преимущество такой организации в том, что она позволяет *Kodi* просканировать и идентифицировать файлы, загрузив метаданные и обложки из Интернета, и получить медиа-библиотеку с полезной — и распознаваемой — информацией, благодаря которой к файлам будет удобно обращаться с других устройств. Подробности о структурировании файлов ищите в вики *Kodi* (<http://bit.ly/XMBCNaming-Files>). Так, для правильного сканирования фильмов имена файлов должны включать только название и год (не обязателен), а за ними должно идти расширение. К счастью, благодаря *File-Bot* (см. стр. 35) переименование файлов не станет каторгой. Заодно можно добавить полезные сведения (скажем, названия эпизодов) к каждому файлу, чтобы лучше их распознавать. В нашем пошаговом руководстве (стр. 35) описан процесс переименования записей телепрограмм, применимый также к фильмам и музыке.

Дав всем файлам правильные имена и сохранив их в правильных каталогах, беритесь за настройку самого *Kodi*. Запустите его, если он еще не запущен, и вы увидите главное окно. Если какие-то дополнения помечены как неработающие, выберите Yes [Да], чтобы удалить их. Затем перейдите в раздел System [Система] и выберите Settings [Параметры] для настройки основной функциональности ядра. Выберите Services > UPnP [Сервисы > UPnP] и нажмите Share video and music libraries through UPnP [Открыть доступ к библиотекам музыки и видео через UPnP], чтобы превратить *Kodi* в настоящий медиа-сервер. Затем пройдитесь по остальным разделам и включите возможности, которые могут понадобиться — например, перейдите в настройки web-сервера [Webserver], и вы сможете настроить

администрирование сервера с удаленного компьютера, набрав IP-адрес сервера в браузере, в формате 192.168.x.y:8080. Обязательно установите пароль для такого доступа. Если вам не нравится внешний вид *Kodi*, смените скин. Зайдите в разделе Appearance > Skin [Внешний вид > Скин] в Settings и выберите PM3.HD, если, например, хотите отображать меню вертикально, а не горизонтально. Большинство других параметров влияют на параметры *Kodi* как медиа-центра, а не как медиа-сервера (подробности — на стр. 36).

Свяжем библиотеки вместе

Настал момент истины: *Kodi* настроен и готов к использованию, как и ваши файлы медиа. Свяжите их друг с другом. Для этого создайте библиотеки для различных типов медиа-файлов — например, просто-напросто Movies [Фильмы] и TV Shows [Телепрограммы]; но при большой коллекции вы можете захотеть разбить эти библиотеки на жанры — Movies (Sci-fi) [Фильмы (научно-фантастические)], TV (US comedies) [Телешоу (комедийные, США)] и т.д. — чтобы упростить просмотр коллекции со всех устройств.

Как бы вы ни расположили свои файлы медиа, процедура добавления любой библиотеки одинакова: в главном окне выберите Videos > Files [Видео > Файлы], и увидите всего один вариант действий: Add Videos [Добавить видео]. Нажмите на нем, чтобы добавить свою первую библиотеку, или «источник», как она называется в *Kodi*. Каждая библиотека указывает на один или несколько каталогов. При правильной организации каталогов нужно указать только один — сделайте это через кнопку Browse [Обзор]. Затем укажите подходящее имя библиотеки (например, TV — Comedy [Телевизионные программы — Комедийные]), и вам предложат установить тип содержимого [Set content], чтобы предупредить о нем *Kodi*. Сначала выберите этот тип (Movies [Фильмы], TV [Телепрограммы], Music Videos [Музыкальные видео] или Music [Музыка]) и скрейпер (базу данных, куда *Kodi* будет помещать обложки альбомов и метаданные), а затем задайте необходимые настройки.

После этого ответьте Yes на предложение обновить библиотеку и подождите, пока *Kodi* просканирует каталог и сопоставит его содержимое с выбранным скрейпером. Настроив первую

библиотеку, переходите к следующей — вернитесь в Videos > Files > Add Videos [Видео > Файлы > Добавить видео] и настройте новый источник. Позже мы подробнее рассмотрим, как добавлять музыку в библиотеку (см. стр. 36), а также расскажем о приемах для добавления фотографий.

Расширим Kodi

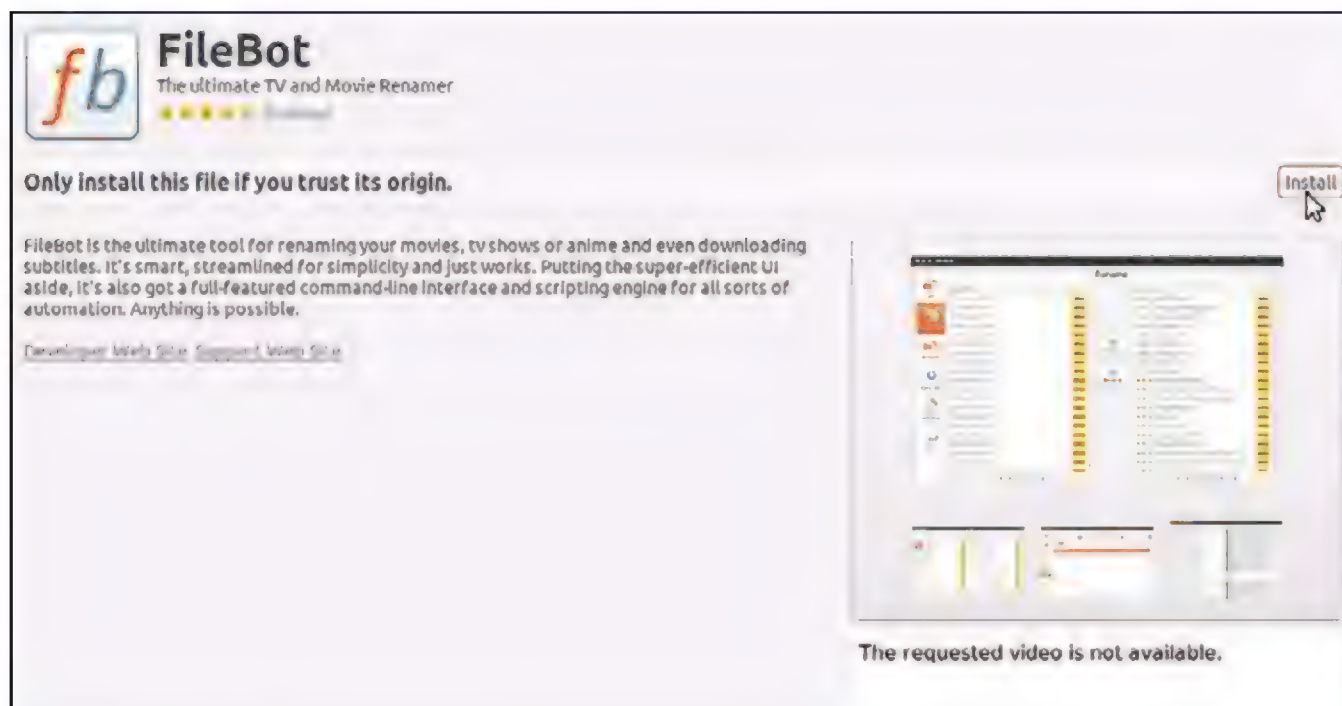
Возможностей *Kodi* вполне достаточно для доступа к медиа-файлам на всех устройствах, на которых файлы читаются, и по локальной сети. Если нужен удаленный доступ к своим медиа-файлам или если у вас есть устройства вроде Chromecast, которые отказываются распознавать некоторые форматы видео, понадобится добавочная помощь. Так как *Kodi* не умеет конвертировать мультимедиа, в отличие от таких альтернатив, как Plex или MediaBrowser, призовем на помощь *BubbleUPnP Server* (<http://bit.ly/BubbleUPnPServer>).

BubbleUPnP работает в паре с уже запущенным сервером, предоставляя отсутствующие в том функции. Он может перекодировать мультимедиа и предоставляет удаленный доступ к медиа-файлам через планшет или телефон с Android, применимый как канал пересылки на другие устройства. Приложение для Android бесплатно, но для преодоления некоторых ограничений придется заплатить £3,04, так что перед покупкой ознакомьтесь с приложением. *BubbleUPnP* достаточно прост в настройке — подробные инструкции можно найти на сайте. Сделав это, откройте браузер на сервере и наберите в адресной строке localhost:5050 для его настройки. Если *Kodi* работает в фоновом режиме, вы должны сразу же его увидеть на вкладке Media Servers [Медиа-серверы].

Имея ТВ-тюнер с интерфейсом USB, почему бы не добавить в список возможностей сервера видеозапись? Тогда вы сможете записывать любимые программы и просматривать их на любом выбранном устройстве позже. *Kodi* поддерживает совместную работу с несколькими ТВ-клиентами, в том числе TVheadend, VDR, MythTV и DVBLINK. Самый известный из них — открытый проект PVR MythTV: он имеется в большинстве репозиториях и обещает работать с любым тюнером, поддерживаемым в Linux. (В будущих статьях мы рассмотрим эту тему подробнее и узнаем, как добавить возможности PVR к своему Linux-компьютеру.)

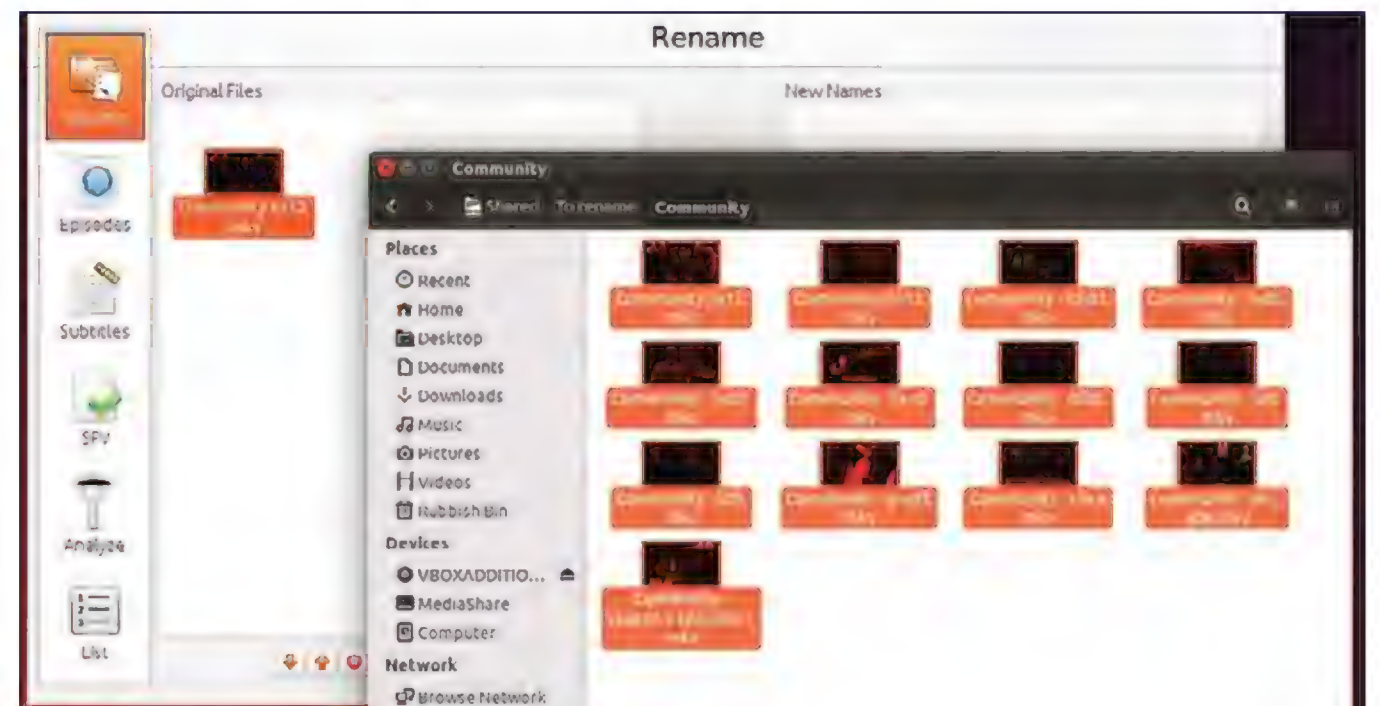
Тип	Структура каталогов	Синтаксис	Пример
Музыка	Music\Исполнитель\Название альбома	Исполнитель — название трека	Music\The Beatles\Past Masters Volume 2\the beatles — обложка writer.mp3
Фильмы	Movies\Жанр\Название фильма	Название (год)	Movies\Sci-Fi\Star Trek\star trek (2009).mkv
Телепрограммы	TV\Жанр\Название\Сезон	Название — номер эпизода в формате sXXeXX	TV\US Comedy\Community\Season 3\community — s03e04.mkv
Музыкальные видео	Music Videos\Исполнитель	Исполнитель — название трека	Music Videos\A-ha\A-ha — did anyone approach you.mkv

Быстро переименовываем медиа-файлы с FileBot



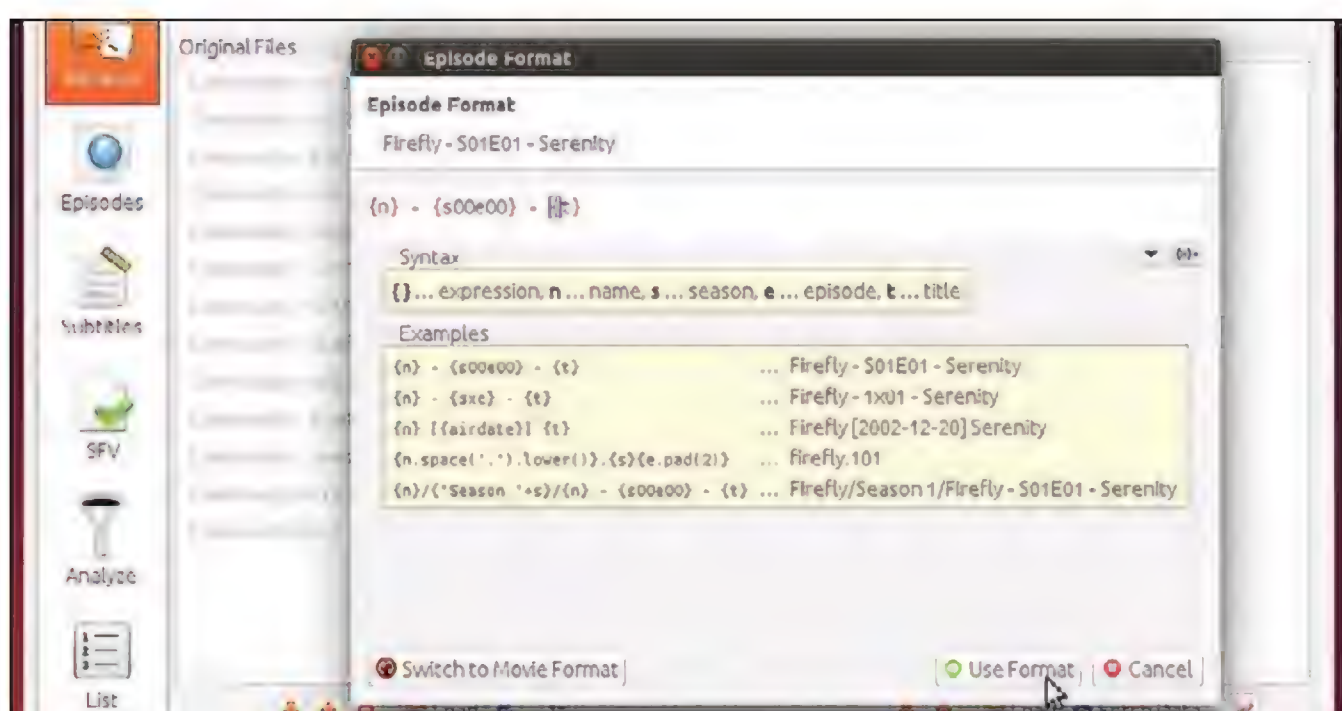
1 Загрузим FileBot

Для использования *FileBot* у вас должна быть установлена Java 8 или более поздней версии. Установив ее, загрузите соответствующий Deb-пакет с сайта <http://filebot.net> и установите его, дважды щелкнув на файле и следуя указаниям мастера (пользователи Ubuntu — за \$4,99 — также могут установить его напрямую из центра приложений Ubuntu). После установки запустите его, набрав “FileBot” в строке поиска или **filebot** в окне терминала.



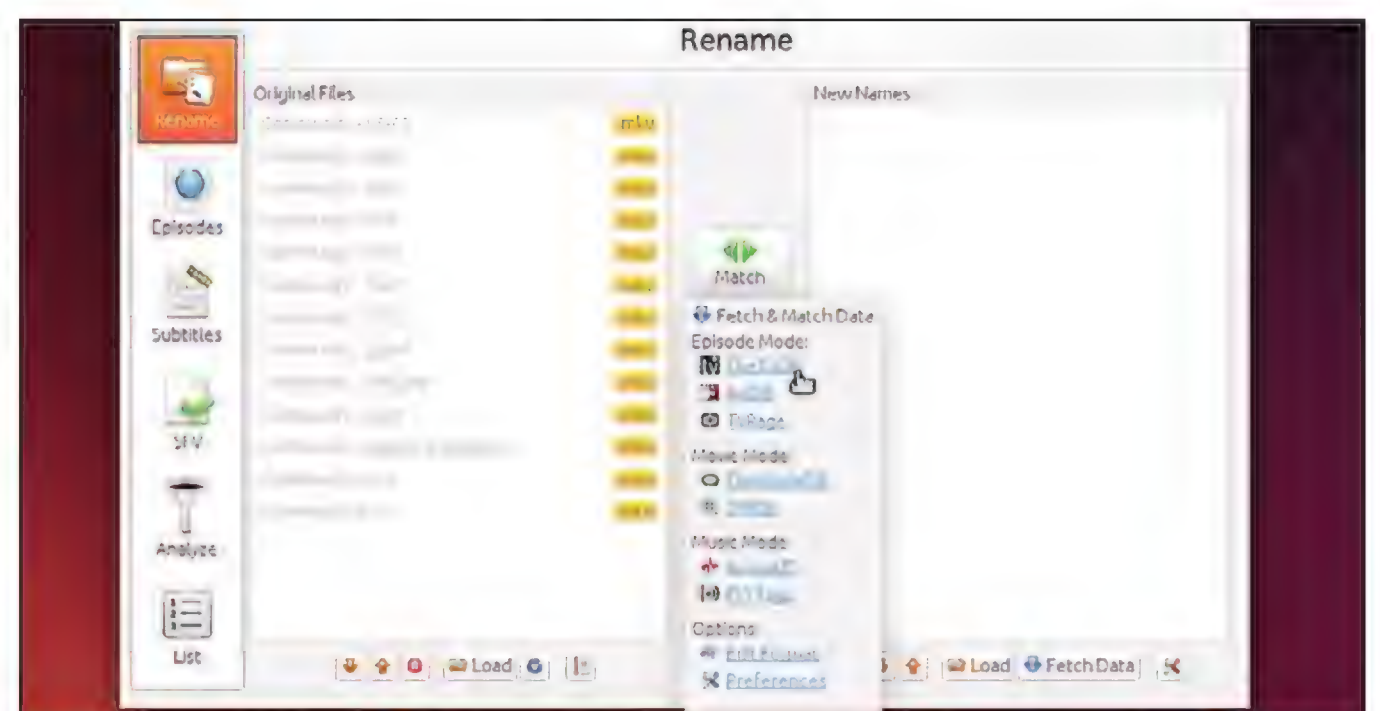
2 Выберем эпизоды

Откройте окно выбора файлов и выберите каталог, содержащий первую телепередачу. Мы советуем переименовывать файлы группами, чтобы процесс был более управляемым, поэтому либо перетащите каталог с программой в окно *FileBot* для выбора всех эпизодов, либо откройте каталог и перетащите в окно каталоги первого или второго сезонов. Выбранные эпизоды появятся в разделе Original Files [Исходные файлы] и будут выделены серым.



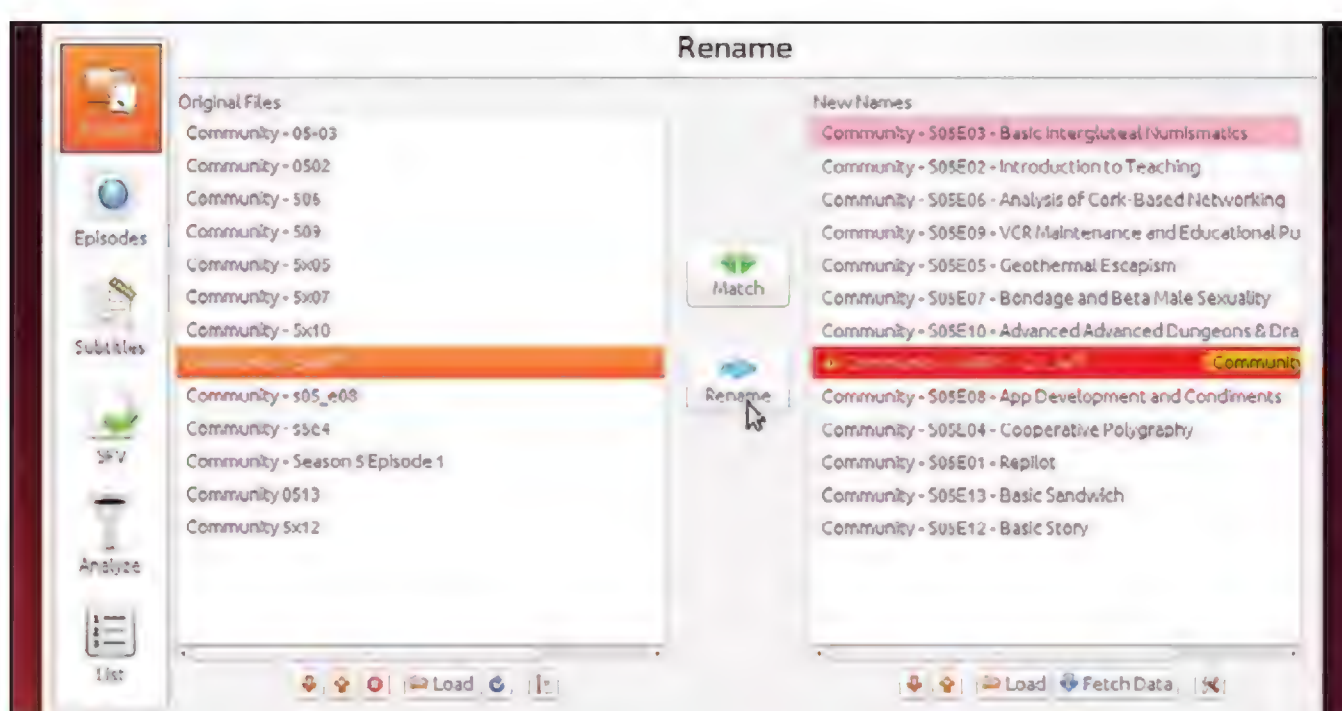
3 Выберем формат именования

Затем нажмите кнопку Match [Соответствие] и Edit Format [Изменить формат] для задания правил именования телепрограмм. Вы увидите ряд примеров — выберите один из них или нажмите кнопку (x)=, чтобы увидеть более популярные «связки», такие как дата эфира или жанр. Чтобы воспользоваться нашим форматом (см. стр. 34) и добавить к нему название эпизода для определения типа файла, наберите в этом поле {n} — {s00e00} — {t} и нажмите Use Format [Использовать формат].



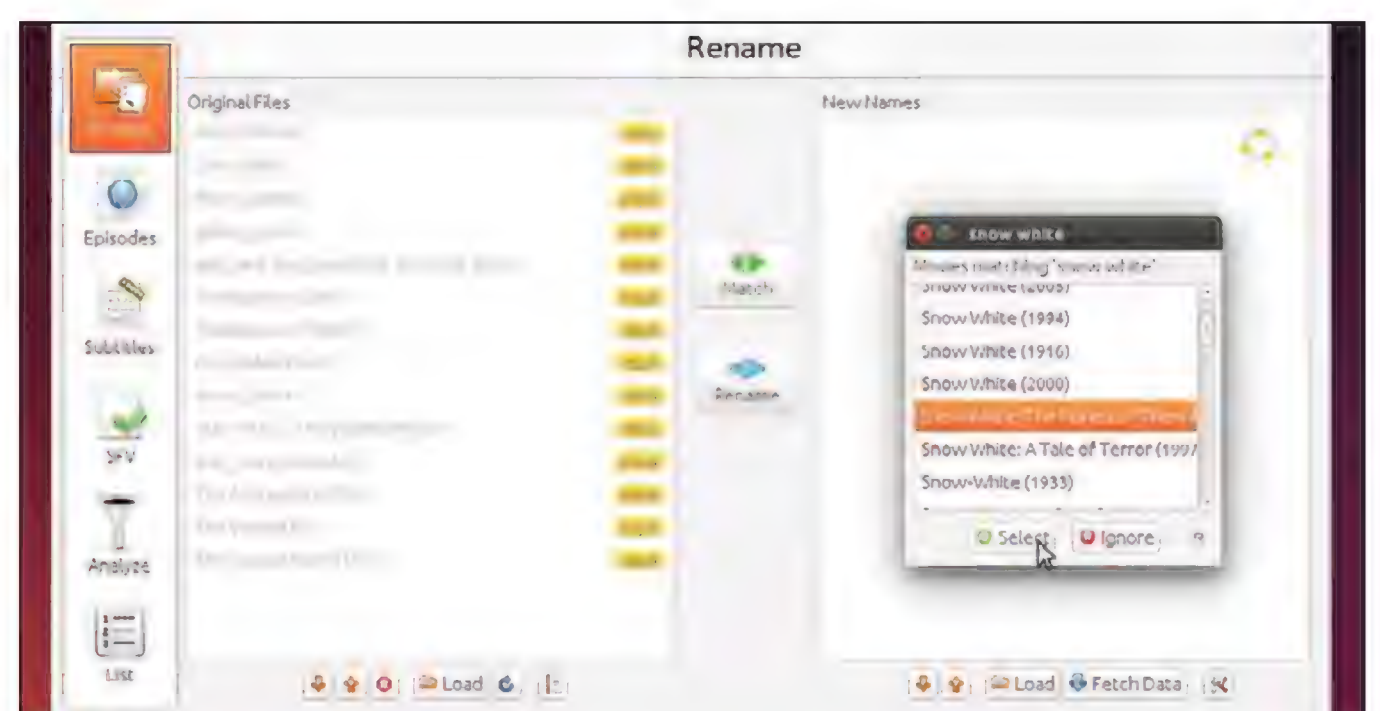
4 Сопоставим эпизоды базе данных

Снова нажмите кнопку Match [Соответствие], но на сей раз выберите одну из трех предложенных баз данных эпизодов [Episode Mode]. Лучший вариант — обычно *TheTVDB*, поэтому начните с нее. В ней будет выполняться поиск соответствия по названию эпизода — если их несколько, перед нажатием Choose [Выбрать] выберите правильное. Затем в окне New Names [Новые имена] появятся предложенные имена файлов.



5 Просмотрим и переименуем файлы

Включение названия эпизода позволит быстро понять, верным ли было соответствие; если нет, попробуйте другую базу данных или другой вариант из предложенных. Насчет соответствий, показанных красным или розовым, *FileBot* не уверен. Если вы всем довольны, нажмите Rename [Переименовать], предварительно удалив все неверные соответствия: выберите эпизод и нажмите кнопку Delete [Удалить].



6 Двигаемся дальше

Если в имени файла есть недопустимые символы, *FileBot* выдаст предупреждение. Для их удаления нажмите Validate [Исправить], и можно переходить к следующей группе эпизодов. Фильмы и музыку также можно переименовать, нажав кнопку Edit Format [Изменить формат] и выбрав Switch to Movie Format [Переключить на формат фильма] или Switch to Music Format [Переключить на формат музыки].



Потоки музыки и фото

Пользуйтесь и делитесь фотками и музыкой из *Kodi* и других источников.

Бiblioteca ваших файлов медиа, очевидно, разбита на три отдельных категории: видео, музыка и фотографии. Возможно, вы уже добавили свою музыкальную библиотеку в *Kodi*, применив совет с предыдущего разворота. При правильной структуре каталогов и имен сделать это просто, и *Kodi* поддерживает широкий диапазон форматов — от вездесущего MP3 до других широко поддерживаемых форматов, от WMA до AAC и FLAC для пуристов.

Музыка добавляется в библиотеку точно так же, как видео: выберите источник, задайте его имя, и после этого *Kodi* просканирует содержимое в библиотеку. Если финальный шаг пропустить, то библиотеку музыки можно будет просматривать только как набор каталогов, а при выборе меню библиотечное представление [Library] будет серым. В таком случае щелкните правой кнопкой мыши на добавленном каталоге и выберите Scan item to library [Сканировать элемент в библиотеку]. На экране должен появиться индикатор сканирования, и по его завершении появится полное библиотечное представление, и вы сможете просматривать содержимое по жанру, исполнителям и другим критериям.

После добавления в библиотеку музыка будет доступна на других устройствах точно так же, как

и другие медиа-файлы (подробная информация для различных устройств приведена на стр. 38). Также можно выводить музыку на различные виды беспроводных колонок. Те, кто пользуется Bluetooth, должны, например, взглянуть на *Blue-man* (<http://launchpad.net/blueman>). Обладатели колонок AirPlay также могут — не напрямую — воспроизводить музыку со своих серверов, но, к сожалению, не с самого *Kodi*.

Потоковая музыка через AirPlay

Если вы хотите воспроизводить музыку на колонки AirPlay, не полагаясь на стороннее устройство вроде мобильного телефона, решение может лежать в звуковой системе *PulseAudio*, имеющейся в большинстве дистрибутивов Linux (но, увы, не в XBMCbuntu). AirPlay не работает с *Kodi* таким образом, но это должно получиться с другими медиа-проигрывателями, в том числе *VLC*, *Amarok*, *Videos* от Gnome и *Rhythmbox*.

Если звук выводится на более старые колонки AirPlay и в Linux установлена *PulseAudio*, то, возможно, у вас уже есть все необходимое: в терминале наберите **paprefs &** и поставьте галочку Make discoverable Apple AirTunes sound devices available locally [Сделать обнаруженные звуковые устройства Apple AirTunes доступными локально]. Нажмите Close [Заккрыть], откройте окно Sound Settings [Звуковые настройки], и ваше устройство AirPlay должно появиться в выходных устройствах. Выберите его и проверьте его работоспособность с помощью *Kodi* или другого музыкального проигрывателя.

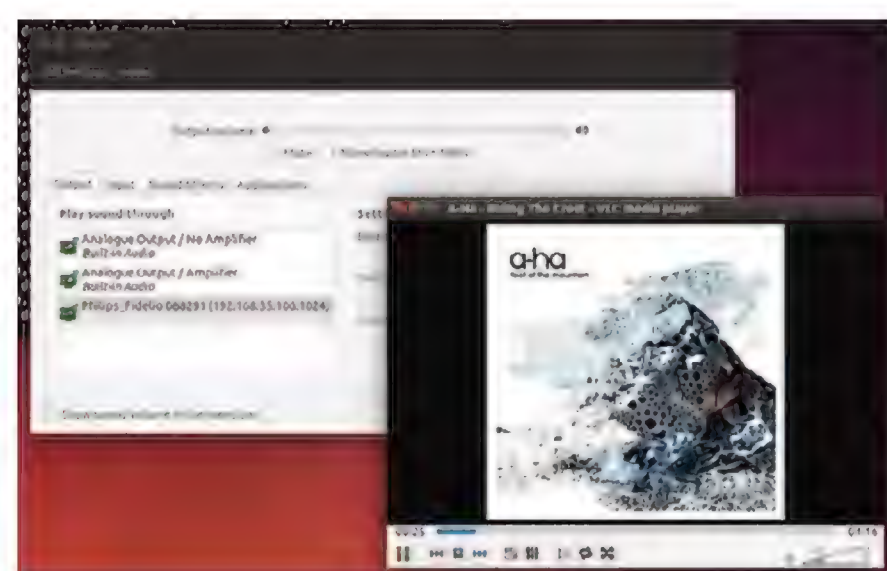
В случае неудачи или при более новом устройстве AirPlay — включая Apple TV 3 — попробуйте проект *PulseAudio RAOP2* (<http://bit.ly/raop2>). На сайте вы найдете инструкции для Ubuntu 14.04, но они подходят и для других дистрибутивов на базе Debian. Воспользуйтесь этими инструкциями для установки предварительных пакетов,

загрузки кода и сборки программы. Затем нужно настроить *PulseAudio*, завершить существующий процесс *PulseAudio*, скопировать конфигурационный файл и запустить новый процесс. Мы успешно проверили это на Philips Fidelio, но ждите большой задержки.

Желая воспроизводить музыку из *Kodi* через колонки AirPlay, можно употребить смартфон как устройство-посредник. При поиске приложения учитывайте, что некоторые бесплатные программы (например, *Streambels* — <http://streambels.com>) одни форматы воспроизводят (обычно MP3), а другие нет (AAC или WMA), поэтому тщательно их протестируйте. Если вы готовы платить за потоковое воспроизведение в свои колонки, то с большинством основных форматов справится *AOA Player Airplay* (<http://bit.ly/AOAPlayerAirplay> — £2,17) для Android и *8player* (<http://bit.ly/8Player> — £2,99) для iOS.

Обмен фотографиями

Стандарты DLNA и UPnP поддерживают обмен фотографиями, а также потоковое воспроизведение музыки и видео по сети. Как ни странно, поддержка UPnP в *Kodi* охватывает видео и музыку, но не фотографии, поэтому придется идти другим путем. Конечно, можно свалить все на плечи *Plex Media Server* или *MediaBrowser* (оба поддерживают обмен фотографиями), но можно и запустить *Plex* вместе с *Kodi* — тогда установите дополнение *PlexXBMC* для доступа к фотографиям, размещенным на *Plex*, из *Kodi*. Загрузите его в виде ZIP-файла (<http://bit.ly/PlexXBMCAddIn>) и установите его в System > Settings > Add-ons [Система > Параметры > Дополнения], вернувшись на верхний уровень и выбрав Install from zip file [Установить из ZIP-файла]. В пошаговом руководстве (см. стр. 37) описано, как добавить фотографии в библиотеку *Plex*. К этим фотографиям можно обращаться через любой из клиентов, описанных на противоположной странице.



➤ Малость безвкусно, но транслировать музыку с вашего сервера в колонки AirPlay вполне возможно.

Kodi как медиа-центр

Основная функция *Kodi* — медиа-центр, но почему бы не возложить на него еще одну задачу, подключив к подходящему телевизору? Для такого использования *Kodi* нужно исследовать его дополнения, реализующие добавочные возможности, такие как подключение к ТВ. Полный список «официальных» плагинов находится на <http://addons.xbmc.org>, но есть и неофициальные репозитории, вроде <http://superrepo.org>, где свыше 1300 дополнений, включая ITV Player. Для добавления нового репозитория в *Kodi* загрузите ZIP-файл репозитория через

браузер. Затем перейдите в *Kodi*, откройте корень раздела Apps [Приложения] в System > Settings [Система > Параметры] и выберите Install from zip file [Установить из ZIP-файла]. После установки нажмите Get Add-ons [Загрузить дополнения], и вы увидите появившийся репозиторий.

Отдельные дополнения также можно загрузить в виде ZIP-файлов — подробный список см. в вики *Kodi* (<http://bit.ly/KodiAddOns>), куда вам и нужно отправиться, если вы хотите, например, установить дополнение BBC iPlayer.

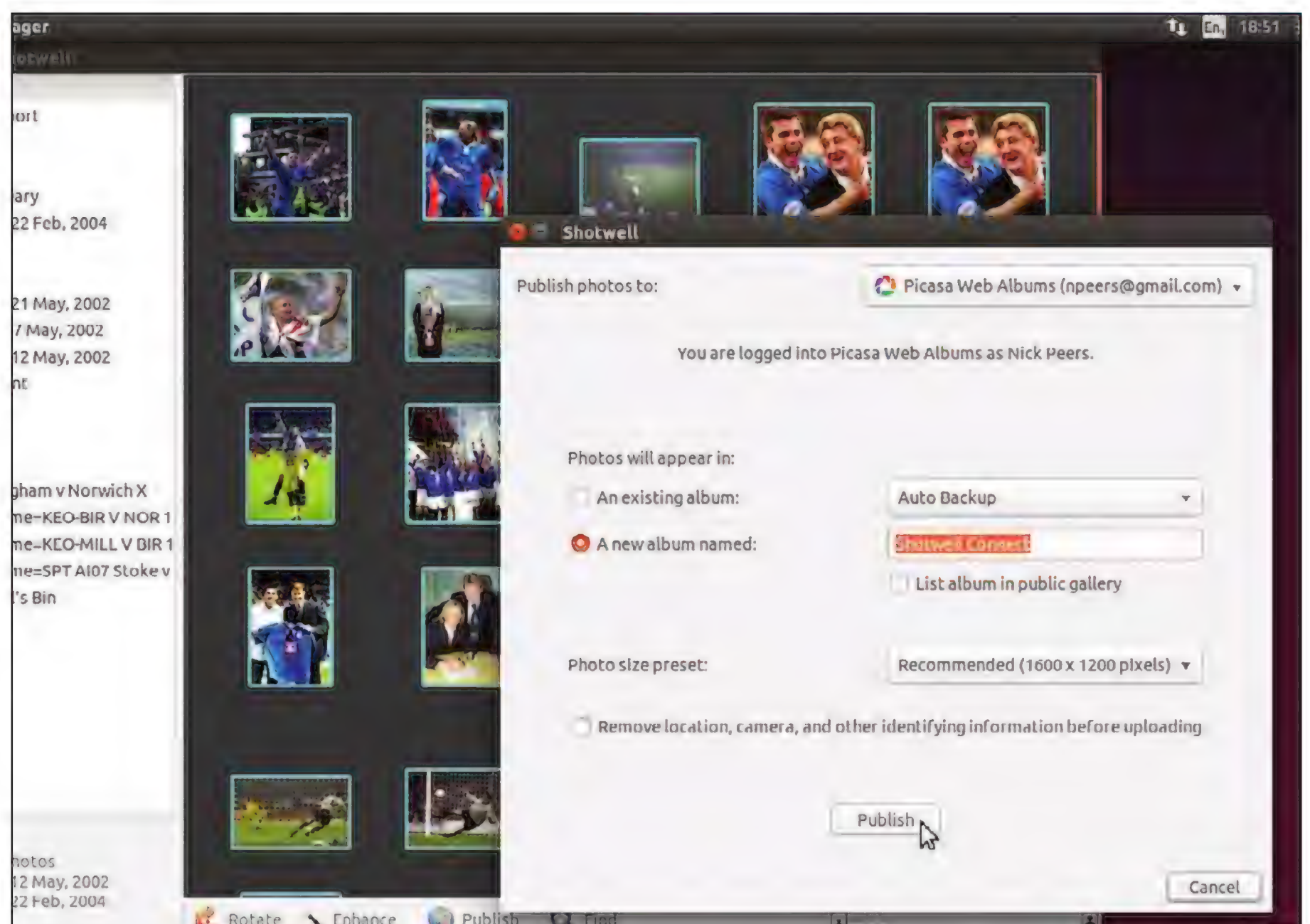


Большинство ваших фотографий (если не все) скорее всего хранятся в Интернете; в этом случае вы уже можете обращаться к ним с разных устройств разными приложениями. Эти приложения не исчерпываются настольным компьютером и мобильными устройствами; большинство популярных сервисов предлагают приложения и для других платформ, например, для Smart TV или проигрывателя Blu-ray. При настройке этих приложений также взгляните на плагин *Pictures* в *Kodi*: он может позволить просматривать ваши фотографии при использовании *Kodi* как медиа-центра.

Google Picasa

Если идея размещения ваших фотографий в Интернете с целью сделать их доступными кажется привлекательной, вам понравится сервис Google Picasa (теперь это часть Google+) (конечно, если вам нравятся продукты Google). Вы бесплатно получите до 15 ГБ (совместно с Google Drive и Gmail) — и он лезет из кожи вон, чтобы сделать ваши фотографии доступными на всяких платформах. Хотя Google недавно официально прекратил поддержку редактора *Picasa* для Linux, его можно установить в *Wine*. К сожалению, загрузка фотографий не работает, поэтому фотографии-то вы подготовите, а загрузить не сможете. А к счастью, подключиться к своей учетной записи Google можно и с помощью других программ. Хороший пример — *Shotwell* (<http://yorba.org>).

Настроив библиотеку, просто выберите файлы, которыми хотите поделиться, и выберите File > Publish [Файл > Опубликовать]. Выберите свою учетную запись Google, затем нажмите Login [Войти в систему], чтобы связать учетную запись с *Shotwell Connect*. Здесь можно выбрать альбом, куда загружать фотографии, а также размер пресета для этих фотографий, и перед загрузкой



» Для загрузки фотографий со своего сервера в учетную запись Google воспользуйтесь *Shotwell*.

указать, стоит ли включать идентифицирующую информацию. Наконец, нажмите Publish, и ваши фотографии будут загружены и автоматически опубликованы.

Кстати, к web-альбомам Picasa можно обращаться с других устройств — например, открыть браузер на настольном компьютере и набрать <https://plus.google.com/photos>. Если у вас есть планшет или телефон с Android, то вы уже знаете, что все снимки камеры автоматически загружаются в Picasa и ими можно управлять с помощью встроенного приложения *Gallery*. Если вы ищете что-нибудь посложнее, попробуйте приложение

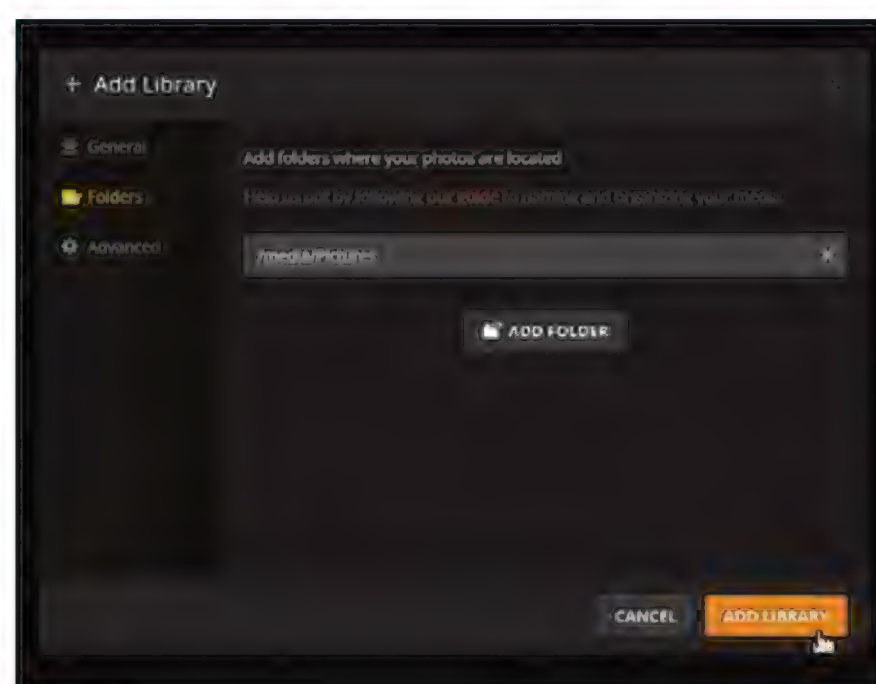
с неуклюжим названием *Tool for Picasa, Google+ Photo* (<http://bit.ly/ToolforPicasa>) от Colifer Lab; оно также поддерживает загрузку ваших фотографий в Chromecast. Если у вас iPad или iPhone, советуем вложить £2,49 в приложение *Web Albums* от Pixite (<http://webalbumsapp.com>), которое позволяет автоматически копировать на сервер фотографии с устройства с iOS, а также просматривать ваши web-альбомы Picasa и управлять ими. Оно также поддерживает загрузку фотографий на Apple TV через AirPlay. Затем следует прочесать подключенные устройства на наличие спецприложения Picasa — оно есть, например, на Roku и Now TV. »

Добавляем библиотеку фотографий на Plex Media Server



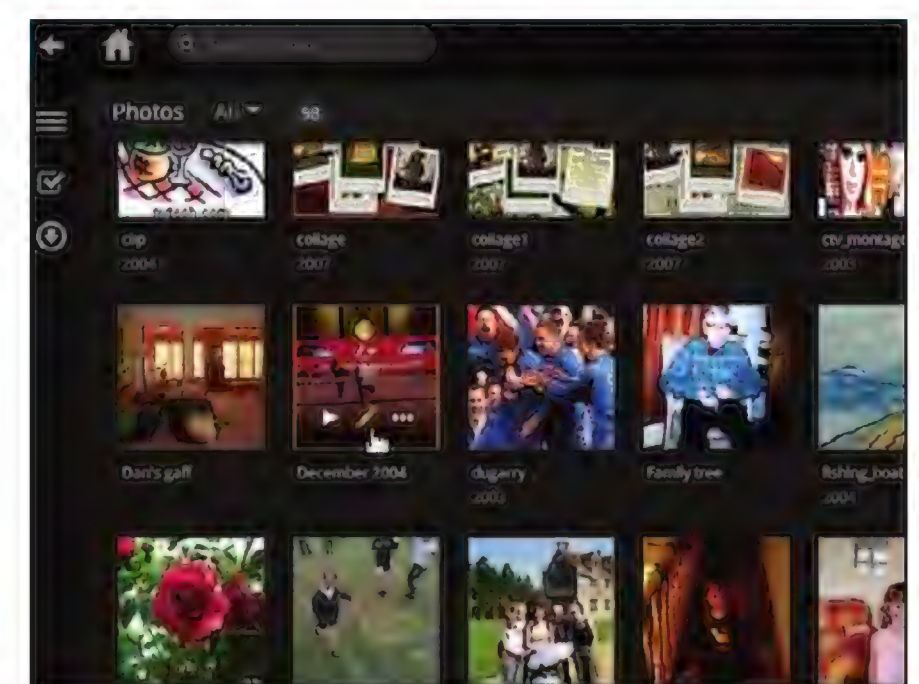
1 Устанавливаем Plex

Не устанавливайте *Plex* из репозитория — скорее всего, там будет устаревший пакет. Лучше загрузите и установите последнюю версию с <https://plex.tv/downloads>. Запустите приложение. Если оно не появилось в браузере, наберите **localhost:32400/web**, чтобы его открыть, затем примите условия соглашения и нажмите кнопку + рядом с именем компьютера для добавления своего первого канала.



2 Добавляем фотографии

Выберите в списке опций Photos [Фотографии] и нажмите Next [Далее]. Нажмите Add Folder [Добавить каталог], чтобы добавить первый каталог с вашими фото — выберите его и нажмите Add [Добавить]. Можете добавить несколько каталогов определенной тематики (например, праздники или семейная история) или главный каталог с фотографиями. Для продолжения нажмите Add Library [Добавить библиотеку]. Добавление в библиотеку потребует времени.



3 Изменяем библиотеку

По умолчанию фотографии сортируются по именам файлов или каталогов. Для изменения имени каталога просто наведите курсор мыши на миниатюру и нажмите на появившуюся иконку карандаша. Здесь можно переименовать каталог, создать название для сортировки [Sort Title] в ином порядке и добавить описание. Для изменения отдельной фотографии выберите ее и опустите под нее курсор мыши, чтобы открылась панель инструментов для редактирования.

Доступ с устройств

Подключаем устройства к вашему новому медиа-серверу.

Медиа-сервер *Kodi* настроен и работает, и теперь вы можете наслаждаться мультимедиа в любой точке дома. Осталось понять, как различные устройства — другие компьютеры, ноутбуки, смартфоны, Smart TV и телевизионные приставки — будут общаться с *Kodi*. Об основах мы уже рассказали, а если вы хотите узнать, как посмотреть сериал «Замедленное развитие» на любом устройстве, читайте дальше.

» **Другие настольные компьютеры и ноутбуки** Очевидный вариант доступа к библиотеке медиа-сервера на другом настольном ПК или ноутбуке — просто установить на него *Kodi*. Сделав это, откройте *Kodi* и выберите Videos > Files > Add Videos [Видео > Файлы > Добавить видео], нажмите Browse и выберите UPnP Devices [Устройства UPnP]. Выберите сервер *Kodi* и нажмите OK, чтобы добавить всю библиотеку как один источник. После этого можно просматривать медиа-файлы точно так же, как на самом сервере — для этого просто выберите сервер из списка доступных каталогов. *Kodi* также может обращаться к содержимому *Plex Media Server* — для максимального удобства работы советуем установить дополнение *PlexXBMC*.

» **Raspberry Pi** Эта популярная маленькая плата — гибкое устройство, и для нее настал звездный час, поэтому мы рекомендуем использовать ее как отдельный медиа-центр для одного из ваших телевизоров. Для доступа к медиа-файлам *Kodi*

удобно воспользоваться приложением *Raspbmc*, которое можно установить на SD-карту в Linux при помощи весьма полезных руководств с www.raspbmc.org. После установки вставьте SD-карту в Pi, загрузите ее для доступа к медиа-файлам и наслаждайтесь.

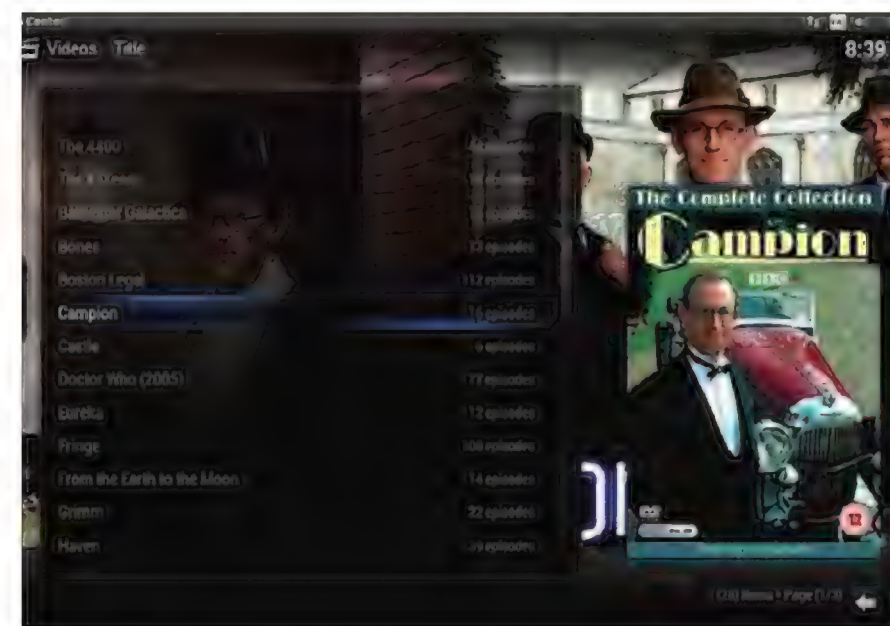
» **Планшеты и телефоны Android** Для Android есть много UPnP-совместимых медиа-проигрывателей, но если вы хотите обращаться к мультимедиа извне домашней сети и нужна возможность перекодирования, потребуется установить *BubbleUPnP* (см. стр. 34). Другой вариант — взять свои медиа-файлы с собой: проигрыватель *VidOn Player* бесплатен (<http://bit.ly/VidOnPlayer>), поддерживает широкий диапазон форматов и позволяет даже загрузить контент для последующего просмотра оффлайн.

Другая альтернатива, если ваш планшет или телефон достаточно мощный — установить официальное приложение *XBMC for Android* — это не *XBMC Remote*, которое в App Store, а полноценная портированная версия самого *XBMC*, доступная для мобильных устройств на Android и практически идентичная настольной версии как внешне, так и функционально.

Перед установкой *XBMC for Android* разрешите в Android устанавливать приложения сторонних разработчиков — нажмите Settings [Параметры], затем Security [Безопасность], прокрутите список вниз и поставьте галочку Unknown sources [Неизвестные источники] в разделе Device

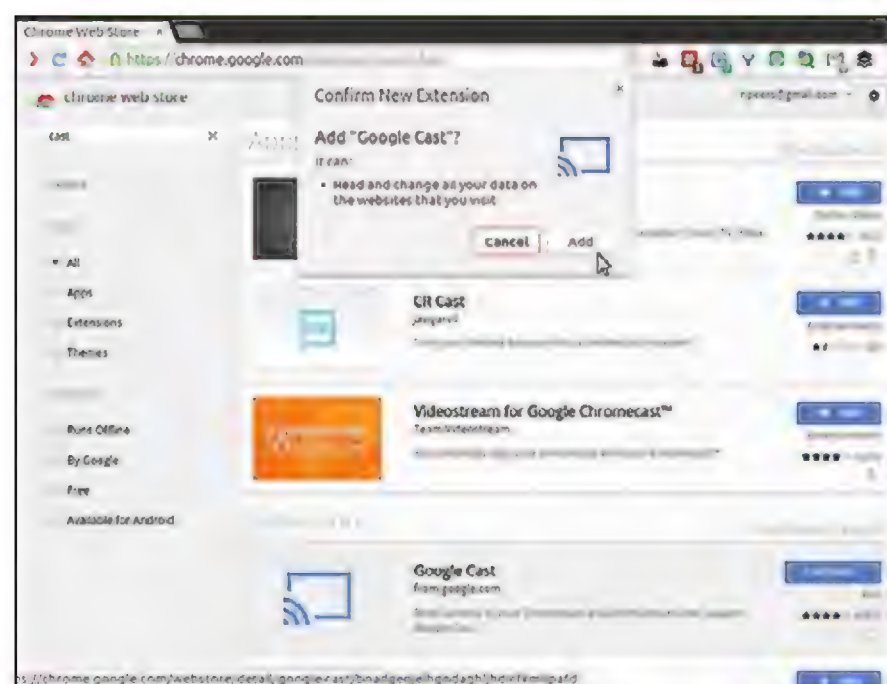
Administrators [Администрирование устройства]. Затем откройте браузер, зайдите на <http://kodi.tv/download> и загрузите подходящую версию для своего устройства (это почти всегда ARM). Загрузка весьма объемистая, так что наберитесь терпения. Когда она закончится, просто щелкните по загруженному файлу и следуйте указаниям мастера установки.

Затем добавьте свой сервер в качестве источника медиа-файлов аналогично настольной версии *Kodi*. Учтите, серверы здесь появляются гораздо медленнее, чем при использовании *Kodi* на настольном компьютере или ноутбуке, так что придется потерпеть. Также будьте готовы к задержкам при просмотре больших файлов — типа фильмов HD-качества — через беспроводную сеть.



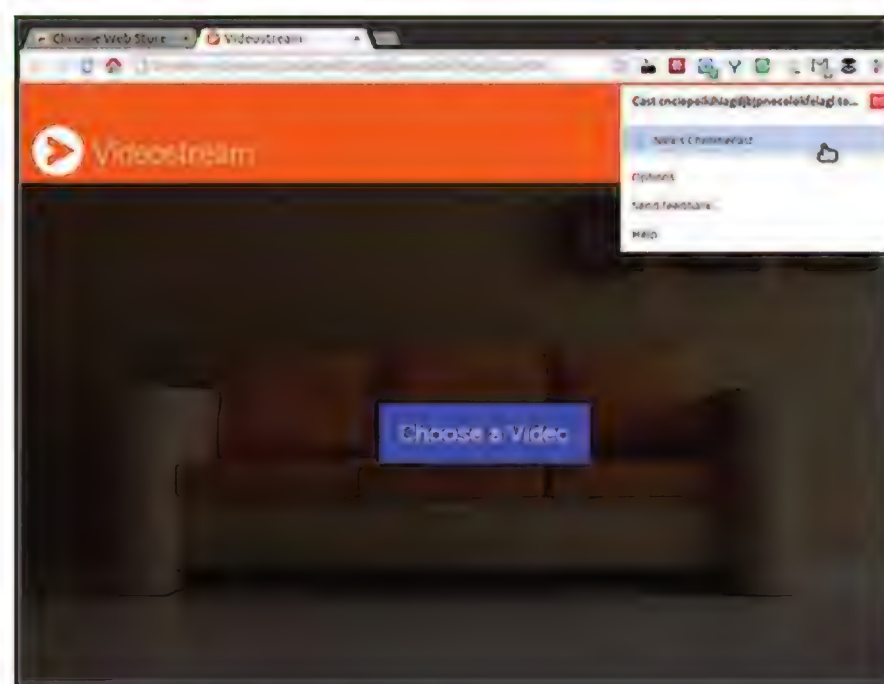
» **Kodi** — прекрасный клиент медиа-центра для установки на настольные ПК и ноутбуки.

Скиньте видео с своего сервера в Google Chromecast



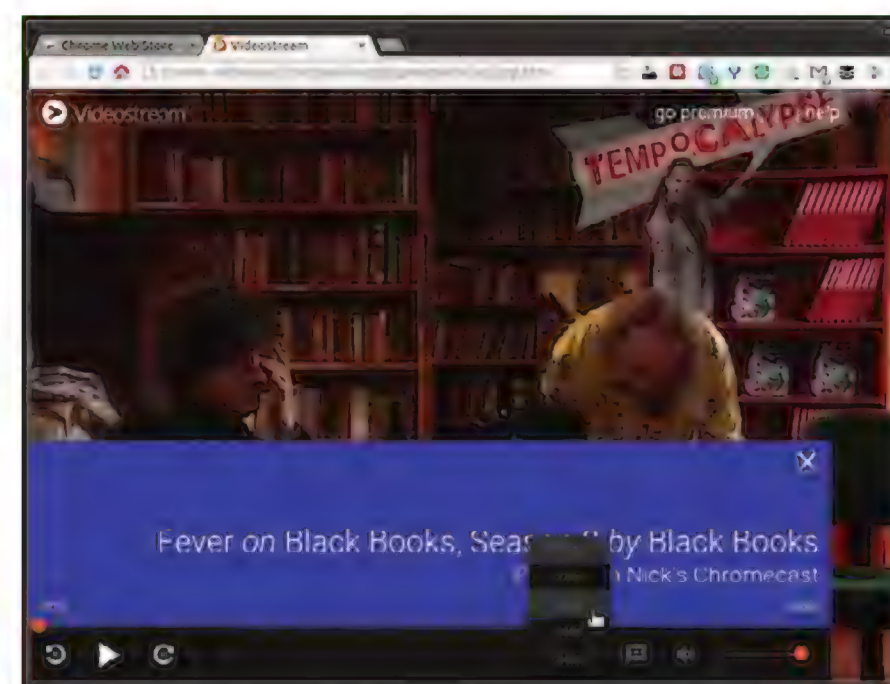
1 Загрузим дополнения

Запустите *Chrome* и зайдите в Chrome Web Store (<http://chrome.google.com/webstore>). В строке поиска по приложениям наберите **Cast** и установите Google Cast, а затем *Videostream for Google Chromecast*. После установки *Videostream* откроется автоматически, а в дальнейшем его можно будет открывать в меню приложений *Chrome* (наберите **chrome://apps** в адресной строке).



2 Выберем устройство

Нажмите кнопку Cast extension на панели инструментов браузера и выберите свое устройство Chromecast. Затем нажмите Choose a Video [Выбрать видео] — помните, *Videostream* не работает напрямую с *Kodi* или сервером UPnP; вместо этого нужно вручную выбрать видео из локального или сетевого хранилища. Выберите файл, откройте его, и файл должен начать проигрываться автоматически в Chromecast.



3 Настроим производительность

Воспроизведение управляется из окна браузера — если качество или производительность не на должном уровне, нажмите кнопку Quality [Качество] и выберите более низкое разрешение: иногда помогает. Другие настройки откроются по кнопке Settings [Настройки] справа от кнопки Help [Справка], а помощь при проблемах ищите на <http://community.getvideostream.com>.

» **iPhone и iPad** Версий *XBMС/Kodi* для устройств Apple нет, а значит, вам придется искать сторонний медиа-проигрыватель для своих задач. *Media Link Player Lite* (<http://bit.ly/MediaLinkPlayerLite>) — бесплатный проигрыватель с функциональным — пусть и не слишком впечатляющим — интерфейсом: свое дело он делает.

Также приглядывайте за *VLC for iOS* (<http://bit.ly/VLCForiOS>) — на момент написания статьи в App Store его не было, но разработчики занимались созданием новой версии для iOS 8 с поддержкой последних моделей iPhone, так что скоро приложение появится в App Store. Как и настольный аналог, *VLC for iOS* умеет воспроизводить почти все форматы, а через пункт меню Local Network [Локальная сеть] — подключаться к своему медиа-серверу на *Kodi*.

Еще одно приложение, которое стоит попробовать — *8player lite* (<http://bit.ly/8Player>). Оно поддерживает широкий диапазон видео- и аудиокодексов и имеет достаточно удобный интерфейс для просмотра содержимого сервера *Kodi*. За £2,99 можно выполнить обновление до платной версии и получить возможность удаленного доступа к серверу *Kodi*.

» **Apple TV** Если у вас Apple TV или Apple TV 2 со сломанной защитой, можете установить *Kodi* на него — подробности см. в вики *Kodi* (<http://kodi.wiki>). После установки этот *Kodi* ничем не отличается от прочих. Если вы хотите транслировать мультимедиа на невзломанный Apple TV, нужно поискать способ трансляции контента на Apple TV через AirPlay. При этом телефон по сути является устройством-посредником. Затем найдите подходящее приложение, способное транслировать контент с медиа-сервера, выводя его на Apple TV через AirPlay.

О пользователях iOS здесь, разумеется, хорошо позаботились — поддержка AirPlay встроена

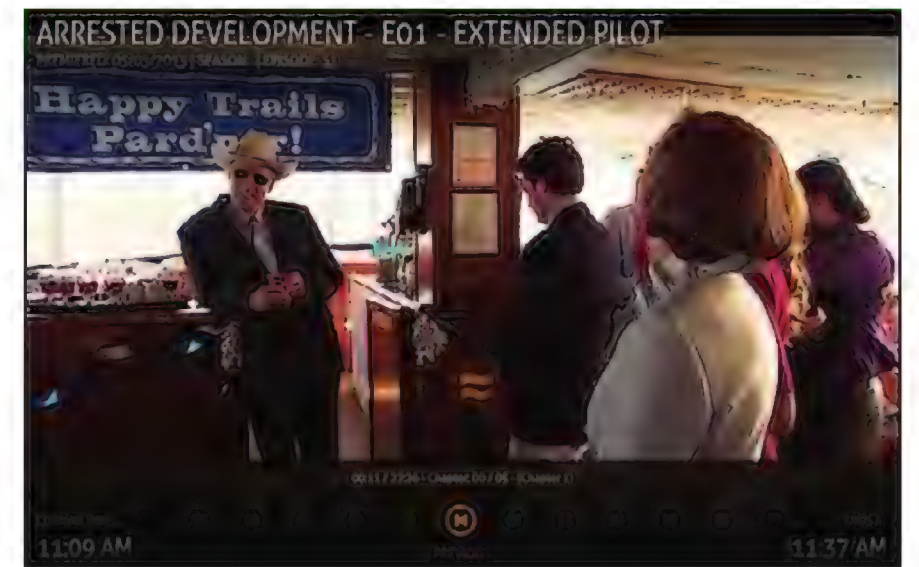
в большинство приложений, включая *Plex*, *MediaBrowser* и *8player*. А хорошие новости для пользователей Android в том, что *VidOn Player* предоставляет поддержку AirPlay и для Android. При использовании *VidOn Player* нажмите Settings [Параметры], затем AirPlay, и переключитесь в режим совместимости [Compatibility Mode] для получения наилучших результатов при трансляции мультимедиа на Apple TV.

» **Smart TV и проигрыватели Blu-ray** Если ваше устройство поддерживает воспроизведение мультимедиа с серверов DLNA/UPnP, вы сможете общаться с сервером, даже сидя в кресле. Большая проблема связана с поддержкой формата: если ваш мультимедиа-файл не закодирован в одном из поддерживаемых устройством форматов, его придется перекодировать.

Сегодня стандартный формат — это кодек x264/H.264 в контейнере MP4 со звуком MP3. Если вы настроили *BubbleUPnP* и у вас устройство на Android, можете перекодировать файлы с его помощью.

» **Roku/Now TV** Медиа-проигрыватель Roku опять же имеет ограниченную поддержку форматов, поэтому не удивляйтесь, если канал MuMedia для доступа к UPnP-контенту не будет работать без помощи *BubbleUPnP*. Между тем урезанный, но зато невероятно дешевый Now TV теперь имеет приложение PlayTo, позволяющее загружать медиа-контент с сервера *Kodi* через устройство Android, на котором запущено соответствующее приложение; но оно требует кропотливой настройки и не очень надежно.

А если у вас *Plex* или *MediaBrowser*, то у Now TV есть один плохо скрываемый секрет, который меняет дело: режим разработчика [Developer Mode]. Включив режим разработчика, вы сможете «подгрузить» одно дополнительное приложение в Now TV через браузер компьютера — в пошаговом



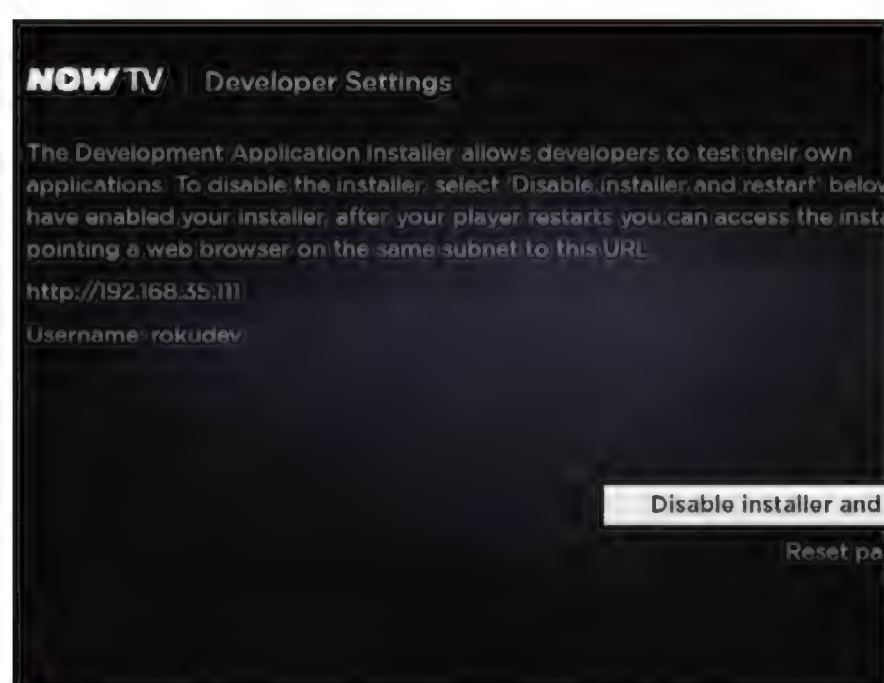
» **XBMС можно запустить на Android точно так же, как и на настольном ПК.**

руководстве внизу описывается, как добавить к Now TV *RARflx* для доступа к серверу *Plex*, и таким же образом можно добавить *MediaBrowser 3* (<http://bit.ly/nowtv-mb3>).

» **Google Chromecast** Здесь поддерживается только видео в форматах WebM и MP4/H.264, поэтому вам понадобится поддержка перекодировки. Самый простой вариант и для Android, и для iOS — это *Plex*, но если у вас Android, перекодировку в желаемый формат также можно выполнить с помощью *BubbleUPnP*.

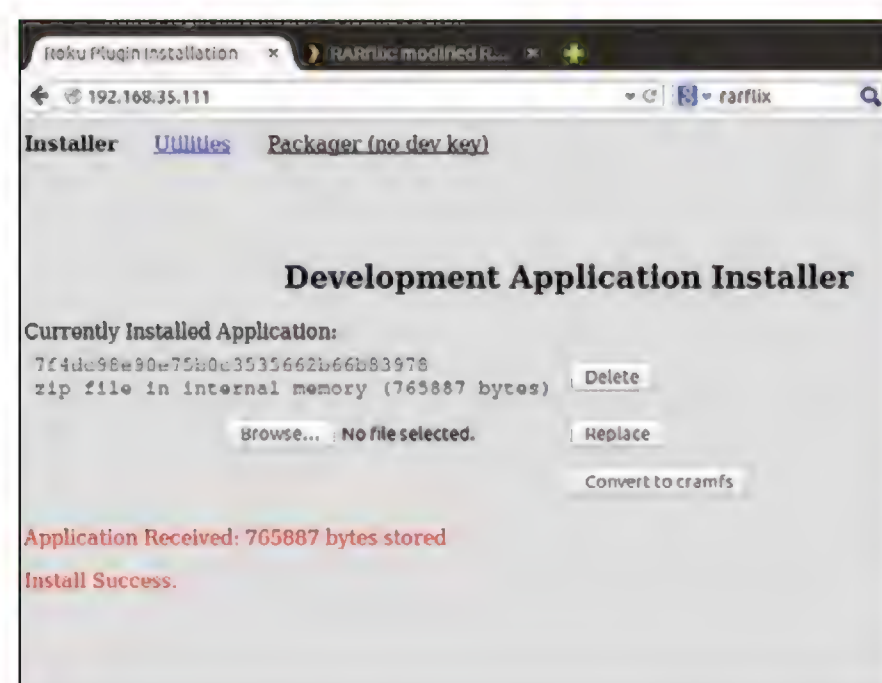
Другой вариант — если вы обращаетесь к *Chrome* (но не *Chromium*) через медиа-сервер — установить плагин *Videostream for Google Chromecast* из Chrome Web Store, который перекодирует и направляет мультимедиа прямо на устройство Chromecast. Это решение менее элегантно, чем прочие, поскольку вам придется выбирать файл, а не просматривать медиа-библиотеку; но в пошаговом руководстве (см. стр. 38) описано все, что вам требуется знать. И если вам нравится Chromecast, но не нравится, что это элемент Google, следите за финансируемым пользователями проектом Matchstick (www.matchstick.tv), который имеет похожую концепцию, но основан на Firefox OS от Mozilla. **LXF**

Доступ к медиа-серверу Plex через приставку Now TV



1 Режим разработчика

На пульте Now TV нажмите кнопку Home трижды, а затем стрелки: 2 раза вверх, вправо, влево, вправо, влево и наконец снова вправо. Появится окно, похожее на рис. выше — выберите Enable installer and restart [Включить установщик и перезапустить], прочтите лицензионное соглашение, нажмите I agree [Принимаю] и снова выберите Enable installer and restart. После перезагрузки задайте пароль доступа к Now TV.



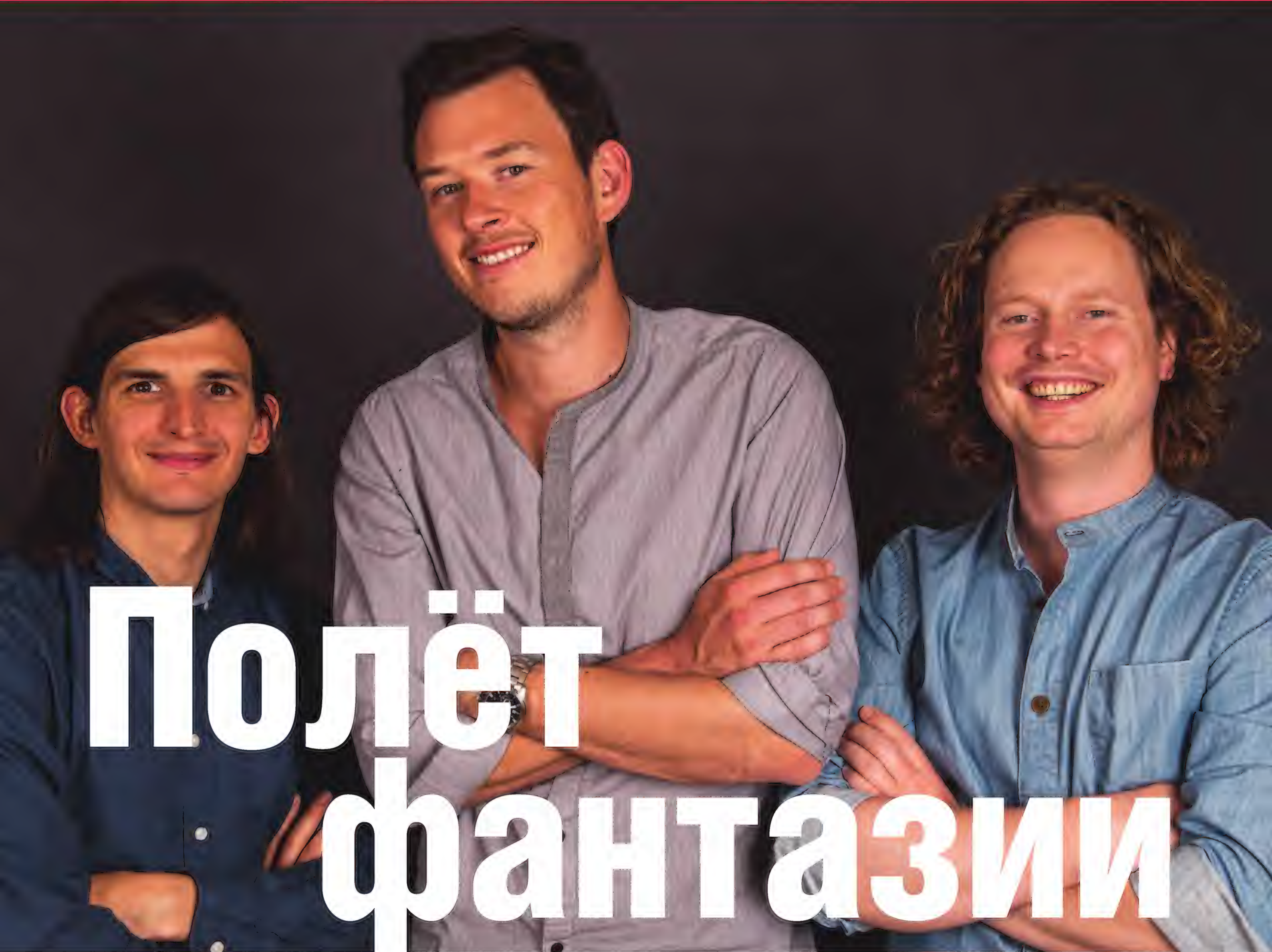
2 Загрузка и установка

Откройте браузер и введите IP-адрес Now TV. Для входа да введите логин **rokudev** и ранее заданный пароль. Теперь откройте отдельную вкладку и перейдите на <http://bit.ly/rarflx-nowtv>, прокрутите содержимое окна вниз и щелкните по ссылке для загрузки *RARflx*, сохранив ZIP-файл в каталог с загрузками. Вернитесь на исходную вкладку, нажмите Browse для выбора ZIP-файла и затем Install [Установить].



3 Наслаждаемся мультимедиа

После установки прочтите приветственное сообщение на Now TV. После этого *RARflx* сразу же найдет медиа-сервер *Plex*, и вы получите доступ ко всему своему медиа-контенту и всем дополнительным web-каналам, которые настроили на своем Now TV за £10. В отличие от официального приложения *Plex* для Roku, *RARflx* полностью бесплатен (хотя пожертвования всегда приветствуются).



Полёт фантазии

Рассел Барнс беседует с небольшой командой шведских предпринимателей — создателей Crazyflie 2.0, крошечной открытой платформы квадрокоптера-для-разработки, который поможет вашим изобретениям взмыть в небеса.



Как вам идея открытой платформы, способной летать? Тобиасу Антосону [Tobias Antosson], Маркусу Элиассону [Marcus Eliasson] и Арно Таффанелю [Arnaud Taffanel] она по душе, и они совершенствуют её с 2009 года. Добившись успеха со своим первым наноквадрокоптером размером с ладонь, эта троица — вместе они известны как Bitcraze — бросила свои прежние профессии, полностью посвятив себя разработке Crazyflie 2.0 размерами 92 × 92 × 92 мм. Мы встретились с ними, чтобы узнать, на кой FLOSS нам трубить про девайс ценой \$180.

Linux Format: Как возникла идея создать крошечный квадрокоптер с открытым кодом?

Тобиас Антоссон: Заскучали на основной работе [смеется]. Мы работали консультантами, и так вышло, что нас всех нанял один клиент. В той компании была такая система: ты выкатываешь идею проекта, а они предоставляют спонсорскую поддержку. Закупают оборудование и всё такое. Причём в других подобных организациях все идеи автоматически становятся

собственностью нанимателя, а здесь это право сохранялось за тобой. Проектировать мы начали в конце 2009 года по вечерам, и уже через полгода взлетел наш первый прототип.

Маркус Элиассон: Это было как раз начало эры квадрокоптеров. Но маленьких ещё не было. На самом деле, мы сначала тоже планировали создать модель обычного размера, опробовать технологию. А потом решили попробовать нечто оригинальное и сделать её как можно меньше — так и родился Crazyflie 1.0

Арно Таффанель: Все мы — специалисты по электронике, и стремились к максимальной простоте. В первой версии моторы были приклеены прямо к печатной плате — то есть мы просто хотели заставить её летать.

LXF: Превратить плату в шасси для квадрокоптера — с лопастями и прочим — отличная задумка. Какие сложности вы преодолели, чтобы она полетела?

ТА: Оглядываясь назад, мне представляется, что с первым Crazyflie дело тормозилось тем, что тогда ещё не было 3D-принтеров в широком доступе. Двигатели на плату мы клеили вручную, и после каждого падения это



приходилось переделывать заново. Отлаживать в таких условиях тяжело. Сначала Crazyflie вообще летал плохо, и на поиск простейших ошибок уходила вечность: он падал, мы годами его чинили — и он снова падал. Пришлось потрудиться, прежде чем мы нашли рабочее решение: приклеили соломинку для коктейля, чтобы устройство держалось в воздухе подольше, и мы смогли найти ошибки.

LXF: Помимо создания одного из первых в мире наноквадрокоптеров, вы решили ещё усложнить себе эту задачу, применяя исключительно открытые инструменты. Почему это было так важно для вас?

ТА: Думаю, в какой-то мере нам было просто любопытно, можно ли создать подобное сложное устройство только с помощью открытых инструментов. Когда мы начинали в 2009-м, приложения для разработки печатных плат вроде *KiCad* (www.kicad-pcb.org) были значительно примитивнее, чем теперь. Исправления ошибок и всё прочее давалось куда труднее.

МЭ: Мы всегда были фанатами Open Source, [поэтому] цель была создать именно открытый проект. А поскольку это ещё и открытая платформа для разработки, очень важно, чтобы нашим пользователям были доступны те же инструменты, что и нам. Если мы при её проектировании могли использовать проприетарные средства, это вовсе не значит, что они были бы доступны и всем членам сообщества. А с приложениями вроде *KiCad* каждый может заглянуть в проектные файлы и поэкспериментировать с ними.

АТ: Вдобавок это позволило создать виртуальную машину со всей средой разработки. Дизайн встроенных систем — дело очень трудное, даже при наилучших условиях: нужно установить все

инструменты, компиляторы и кросс-компиляторы. Поскольку у нас всё это открытое, нам удалось сделать виртуальную машину Ubuntu, где есть все необходимые инструменты и предустановлено

ПРО CRAZYFLIE 1.0

«В первоначальной версии двигателя были приклеены прямо к плате.»

ПО Crazyflie. Она легко и быстро скачивается и работает практически на любом компьютере — что с проприетарным инструментарием было бы вообще невозможно.

LXF: Сейчас готовится к выходу вторая версия вашего квадрокоптера. Будет ли 2.0 всецело ориентирована на исправление ошибок первоначальной версии, или вы задумали какие-то нововведения?

ТА: Некоторые элементы новой модели призваны исправить недостатки исходного дизайна, но главной целью стало создание более функциональной летающей платформы, для разных способов применения. Интерес к разработке дронов растёт, так что универсальность стала очень важна.

МЭ: В первую очередь мы решили добавить порт расширения. Именно на него были

направлены все незадействованные ресурсы процессора первого Crazyflie, чтобы люди могли подключать к нему всё, что угодно. И наши пользователи пробовали и новые устройства, и датчики, но дело шло тяжело. Приходилось припаивать [оборудование] и самостоятельно писать драйверы — это было не так просто.

В Crazyflie 2.0 мы улучшили этот порт — к нему стало можно с ходу подключить гораздо больше »



» Дополнение в виде кольца светодиодов превращает Crazyflie 2.0 в летающее блюдце.

всякой всячины. Мы также планируем продавать платы расширения, подключив которые, пользователи смогут запросто расширить возможности своих платформ. Кроме того, они могут создавать собственные такие платы, и мы стремимся им это упростить.

LXF: Crazyflie 2.0 умещается на ладони. Какой вес он выдерживает, не рухнет?

ТА: Как раз грузоподъёмность стала одним из усовершенствований в Crazyflie 2.0. Она увеличилась с 5 до 15 грамм. Не так уж много, но более чем достаточно, чтобы выдержать небольшую камеру или устройство GPS. Благодаря этому мы сделали побольше моторы и аккумулятор, тем самым улучшив лётные характеристики.

LXF: Какие ещё новые функции есть в Crazyflie 2.0?

МЭ: Мы добавили спящий режим — в нём новая модель может находиться до года, прежде чем разрядится аккумулятор. Может, для тех, кому Crazyflie только игрушка, это не особо интересно, но можно распределить по определённой территории несколько квадрокоптеров, чтобы они поочередно просыпались и выполняли некие задания. Мы старались впихнуть в устройство как можно больше функций, так что простор для инноваций необычайно широк.

МЭ: Мы также добавили [поддержку] Bluetooth с низким энергопотреблением. Через Bluetooth LE можно управлять Crazyflie 2.0 с устройства Android или iPhone.

АТ: Bluetooth LE нужен, главным образом, чтобы облегчить знакомство с Crazyflie тем, для кого это просто развлечение. У нас уже есть рабочие прототипы приложений для управления Crazyflie на базе Android и iOS, и вскоре они появятся на рынке.

МЭ: А ещё у нас есть новый Crazyradio. Crazyradio PA — это USB-модем, с помощью которого можно

управлять Crazyflie с вашего ПК, используя контролер Xbox или PlayStation. Он ловит в радиусе до километра в пределах прямой видимости и при хороших погодных условиях. Прежнюю версию Crazyradio тоже никто не отменял.

LXF: Помимо управления Crazyflie с ПК, какие ещё возможности даёт Crazyradio PA?

МЭ: Это также блестящий способ получить низковольтный доступ с низкими задержками к самому Crazyflie. Протокол по USB-модему, как и протокол для радио, полностью задокументированы в нашем wiki.

Используя Crazyflie со своего компьютера, вы получаете доступ к настраиваемым программам управления полётами, а также многим API для разработки. Мы используем Python, но есть и API для C, C++, Java, JavaScript, Ruby и так далее. Всё это работает на Mac, Windows и Linux.

LXF: Кто является потенциальными покупателями Crazyflie 2.0?

МЭ: Одна из ключевых проблем состоит в том, что потенциальный рынок сбыта очень широк: для кого-то это просто игрушка, для кого-то — платформа разработки или инструмент исследования.

В качестве игрушки он выдаёт море удовольствия прямо из коробки. Нужно только распаковать его и за пару минут собрать, а потом приказывать полететь — с вашего устройства iPhone или Android. В отличие от первого Crazyflie, ничего пахать не нужно, что очень порадует рядовых пользователей. Летаёт он здорово, а производительность безумная.

Когда вам надоедят его полёты вокруг вашего дома, настанет черёд проектирования и разработки, которые вы будете постепенно осваивать. По сути, это летучая платформа для разработки.

С другого края спектра, мы видим, что Crazyflie применяют учёные. Их интересует не столько

управление им с помощью iPhone через Bluetooth LE... сколько то, какие открытые библиотеки в нём поддерживаются.

Возможности Crazyflie как открытой платформы для разработки невероятны. Можно заниматься iPhone, Android, встроенным C, Python... можно даже создавать собственные API для вашего ПК. Для любого, кто хочет поэкспериментировать с летающей робототехникой, это великолепная платформа, способная воплотить всё, что вам интересно.

LXF: Как он летает? К первой модели надо было привыкнуть.

ТА: Я понимаю, что это прозвучит нахально, но даже трудно выразить, как классно он летает и как здорово ощущать, что он под твоим контролем. Невероятная скорость и точность — а побочным эффектом [увеличенной] грузоподъёмности стала большая манёвренность: он может двигаться с наклоном до 40 градусов, не меняя направления, и почти мгновенно исчезает из вида.

МЭ: Надо совершить четыре-пять полётов, чтобы полностью с ним освоиться, но это [количество] зависит от самого человека. У тех, кто моложе 20, он ударится в потолок раньше, чем вы успеете объяснить, зачем нужны кнопки на контроллере; а пользователи постарше будут гораздо осторожнее. Как бы мы ни старались, пятнадцатилетки, поднатеревшие в играх на PlayStation, нас посрамляют [смеется].

АТ: Все параметры управления можно настроить через программу на ПК. Новички, например, примут не столь радикальные настройки, как мы. Можно уменьшать тягу, скорость вращения, наклон.

С первым Crazyflie всё время был риск, что он упадёт. Crazyflie 2.0 гораздо надёжнее, с ним уже не так этого боишься. Новые монтажные опоры двигателей обеспечивают хорошую защиту — теперь он просто подпрыгивает.

LXF: А как работает порт расширения? Смогут пользователи купить платы расширения отдельно?

ТА: На порт расширения мы в начале разработки потратили много времени. Мы хотели, чтобы добавочные платы в Crazyflie 2.0 можно было крепить и снизу, и сверху. Нам нужна была система, позволяющая прикреплять бесчисленное количество плат, хотя, конечно, фактор веса и баланса вскоре также пришлось учитывать.

Мы перебрали множество систем и коннекторов и в конце концов нашли решение, разместив дополнительные платы над аккумулятором. Это одна из главных функциональных особенностей новой модели — возможность адаптировать Crazyflie под свои нужды.

Из запланированных нами расширений, уже есть беспроводной зарядник, на той же технологии, что и во многих современных смартфонах, а также кольцо светодиодов со множеством RGB-лампочек, прикреплённое к основанию Crazyflie и направленное вниз. При слабом освещении это создаёт потрясающие эффекты.



Команда Bitcraze. Слева направо: Тобиас Антоссон, Маркус Элиассон, Арно Таффанель.

LXF: А как насчёт создания собственных плат расширения?

ТА: У нас есть отдельная подключаемая плата, которая позволяет прикрепить Crazyflie к плате прототипирования, чтобы испытать собственные варианты расширений. И есть макетная плата расширения, к которой можно припаять ваше оборудование и запустить свою идею в полёт.

МЭ: Мы, к примеру, для демонстрации макетной платы изготовили летающий светофор. Припаяли три цветных диода друг над другом, в форме светофора, и написали для них программу управления.

Это очень примитивный пример, но в этом духе можно сделать что угодно. Можно создать маленький летучий динамик, припаяв к макетной плате пьезоусилитель, и летать, досажая соседям всякими шумами. У нас есть и другие идеи в разработке — плата для GPS и прототип для камеры. Обдумываем также плату с SD-картой, чтобы записывать большие объёмы данных.

LXF: Делаете ли вы что-нибудь в поддержку разработки плат сообществом?

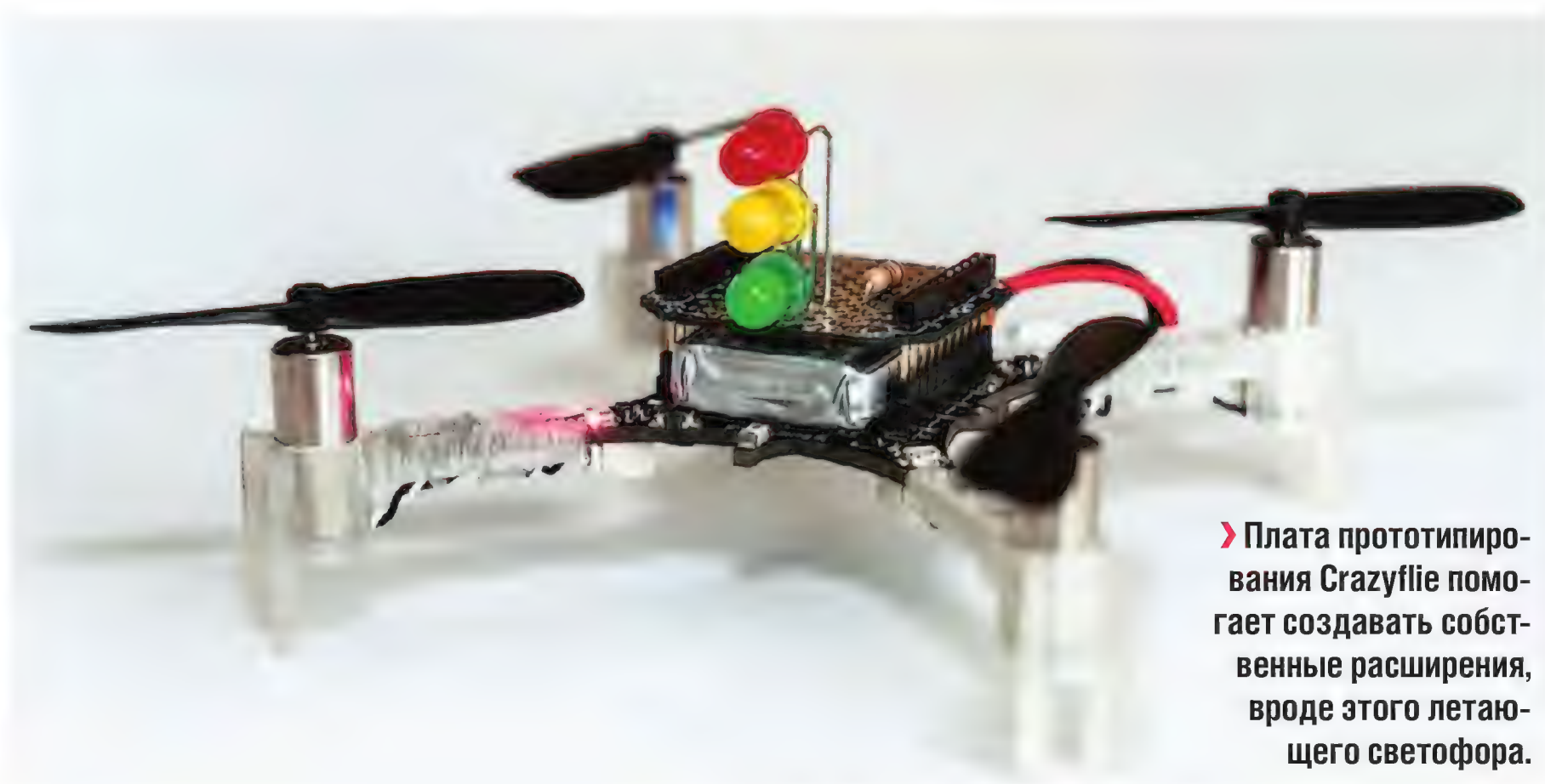
ТА: Мы очень надеемся, что сообщество поделится с нами множеством идей. Мы сделали шаблон для создания собственных платформ расширений, который добавим и в нашу виртуальную машину для разработки.

МЭ: Если вы откроете проект в *KiCad*, там уже будут компоненты схемы коннектора, так что через библиотеки *KiCad* вы можете найти и подключить любое устройство. Сами коннекторы мы тоже добавили — вам осталось только подцепить оборудование.

Кроме того, мы обрисовали контур вокруг Crazyflie 2.0, чтобы было понятнее, какой зазор нужен для разъёмов и винтов. Так что вы будете хорошо представлять себе, как будет выглядеть готовая [плата].

LXF: Планируется ли сделать Crazyflie 2.0 способным на автономный полёт?

АТ: Одна из главных сложностей с тем, чтобы заставить штуку вроде Crazyflie 2.0 летать автономно — это точность определения местоположения в трёхмерном пространстве. Используя такие вещи, как Microsoft Kinect, мы разработали самое простое решение, а уже сообщество создало несколько более продвинутых вариантов.



➤ Плата прототипирования Crazyflie помогает создавать собственные расширения, вроде этого летающего светофора.

С помощью Kinect, Crazyflie может летать совершенно автономно — но, естественно, в радиусе действия датчика.

Мы также сотрудничаем с компанией, которая занимается простыми датчиками оптического потока. Так что надеемся, что в будущем станет возможным посылать Crazyflie автоматизированные команды, например, «подняться на метр» или «вперёд на метр».

О БОЛОНСКОМ УНИВЕРСИТЕТЕ

«С нашей платформой создают алгоритмы поиска пострадавших в горах.»

LXF: А как насчёт использования GPS для получения довольно точных данных о местоположении, как в больших квадрокоптерах?

ТА: Безусловно, с GPS это намного проще, но Crazyflie ориентирован на применение внутри помещения. Он будет отлично летать и на улице, но не приспособлен для суровых метеоусловий. То, что с Crazyflie можно работать в помещении, во многом объясняет его популярность среди учёных. Чувствительное лабораторное оборудование должно находиться под контролем.

Так, исследовательская лаборатория в Болонском университете использовала Crazyflie для автономной работы с их очень дорогой системой слежения. С помощью нашей платформы они создают алгоритмы поиска и спасения пострадавших в горных районах.

LXF: Над чем вы работаете прямо сейчас, подготавливая очередной релиз Crazyflie?

АТ: Сейчас главное для нас — архитектура. [Мы хотим] чтобы пользователям было как можно проще создавать собственные программы и использовать порты расширения.

В перспективе, мы хотим, чтобы идею, пришедшую утром, можно было прототипировать, написать весь код и запустить в первый полёт уже к концу дня. Пока это только мечта, но чтобы это стало настолько просто, от нас требуется много усилий.

Нам представляется, что для добавления кода и драйверов новой платы расширения не нужно перебирать весь квадрокоптер. Нужно сделать так, чтобы для этого понадобилось всего один-два файла — над этим мы и работаем.

Нас вдохновил способ работы драйверов Linux и Android. По сути, мы решаем ту же проблему. В Linux, имея дело с драйвером устройства, вы меняете не все файлы, а только один, кото-

рый автоматически регистрируется и может запрашивать доступ к различным ресурсам. Мы хотим, чтобы у нас была система, которая, скажем, распознавалась бы как драйвер расширительной платы

диодного кольца, требующий доступа к PWM [Pulse Width Modulation — широтно-импульсная модуляция].

LXF: Зачастую изготовление — это самая сложная часть проекта, связанного с оборудованием. Как у вас с этим в Crazyflie 2.0?

ТА: Мы сотрудничаем с китайской компанией под названием Seeed Technology (www.seeedstudio.com), которая занимается множеством подобных проектов. В данный момент мы внимательно следим за тем, как они подбирают компоненты, чтобы быть уверенными в качестве.

В конце концов, просто гарантировать, что каждый сошедший с конвейера квадрокоптер Crazyflie 2.0 работает — уже огромная ответственность. Все мы уже побывали в Китае, чтобы самостоятельно испытать первые образцы.

Путешествия с целью первыми всё увидеть самим многое упрощают... во-первых, вести общение куда лучше лично. Да и работать с Seeed очень здорово: у них уже есть опыт взаимодействия с маленькими западными компаниями вроде нашей, так что дела идут лучше, чем могло бы быть. **LXF**

Сейчас Crazyflie 2.0 доступен для предварительного заказа. Поставки должны начаться вскоре после того, как этот номер журнала поступит в продажу. Свежую информацию см. на сайте Bitcraze: www.bitcraze.se.



Самопалим себе NAS

Джонни Бидвелл демонстрирует подход DIY [Сделай Сам] к производительности при создании сетевого хранилища данных.



По мере удешевления хранилищ данных и разрастания аппетита к данным, все больше людей ищут устройства NAS (network-attached storage — сетевое хранилище данных) для складирования битов своей информации. Предлагаются многочисленные модели готовых устройств от множества производителей и по самым разным ценам. Наиболее распространенными являются устройства с двумя или четырьмя дисками, реализующие все функции, необходимые для домашнего хранилища.

Вас вряд ли удивит, что в сердце многих из этих технологий лежит Linux, поскольку это всего-навсего скромные компьютеры x86 или ARM, снабженные интересными

web-интерфейсами и простыми в использовании программами настройки.

В принципе, люди небогатые могут слегка сэкономить, создав собственное устройство NAS. Есть несколько специализированных NAS с открытым кодом, и самое популярное трио — тут не обошлось без кровосмешения — *FreeNAS* на базе BSD (переписанный старый проект

«Вас вряд ли удивит, что в сердце многих из этих технологий лежит Linux.»

с тем же названием), *NAS4Free* (развитие первоначального кода *FreeNAS*) и *OpenMediaVault* (проект на базе Debian от автора *FreeNAS*). Все это —

замечательные проекты, и они дадут вам всё необходимое: от создания массива до выдачи в доступ ваших пиратских записей Криса де Бурга и прочих файлов. Ну, а вдруг вы хотите настроить что-то самостоятельно? Возможно, вам надо, чтобы ваш компьютер NAS служил еще и медиacentром, или отсылал поток игр *Steam* с компьютера с Windows, или запускал *ownCloud*? Или

вы намерены все держать под контролем, особенно в свете нашумевших атак Shellshock на устройства NAS с выходом в Интернет... Короче, давайте выясним, как все это сделать.

Что касается оборудования, важная его часть — жесткие диски. Функции хранилища может выполнять и один большой диск, но лучше приобрести их несколько,

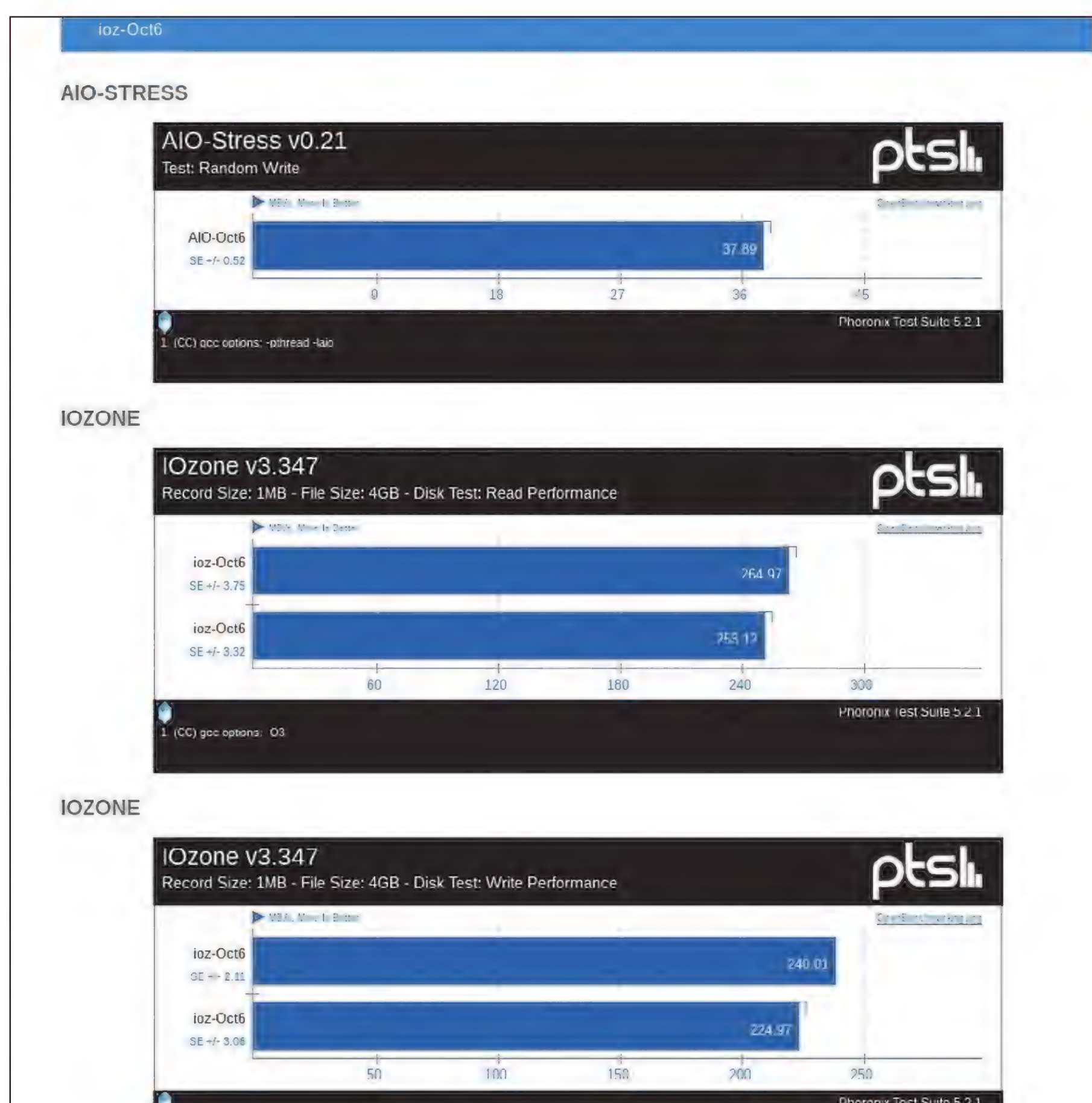
для некоторой избыточности. RAID потребует от двух до шести дисков, и всё будет намного проще и эффективнее, если они будут одного объема. С двумя дисками вы получите конфигурацию RAID1 (где один диск зеркально отражает другой), с тремя дисками у вас будет RAID5 (где данные и блоки четности распределены по дискам для обеспечения повышенной производительности и целостности). Мы выбрали четыре диска, в основном из-за того, что щедрые парни из Western Digital прислали нам четыре диска серии Red по 6 ТБ каждый.

Создание RAID

С четырьмя дисками возможен целый ряд конфигураций RAID, о чем мы расскажем вкратце. Не волнуйтесь — обо всяких тонкостях и экзотических вопросах, связанных с дисками, всё сказано на с. 50. RAID10 — это сочетание уровней 1 и 0, чтобы мы сначала создали массив из двух дисков RAID0 (который не предлагает избыточности, но удваивает производительность), и затем сделали его зеркальное отражение. RAID5 опять же возможен, но не рекомендуется, поскольку при потере одного диска и последующей перестройке массива интенсивный поток операций ввода/вывода заметно увеличит шансы потери еще одного диска — и, следовательно, всех ваших данных. RAID6 предлагает страховку от отказа двух дисков и небольшое увеличение скорости за счет распределения порций данных между дисками, и именно его мы выбрали для нашей структуры. В конечном итоге у нас должно оказаться 12 ТБ полезного пространства и скорость передачи вдвое выше, чем на одном диске.

Можно бы поместить ОС на отдельный раздел одного из ваших дисков RAID, но мы такого не рекомендуем: тогда вам пришлось бы соответственно уменьшить объем всех ваших разделов RAID, ну и вообще неплохо держать такие вещи отдельно. Можно также установить ОС внутри массива, если у загрузчика имеется собственный раздел и ваш образ initrd имеет поддержку *mdadm* (Multiple Disk Administration). И опять же, для хранения это не подходит.

Мы истратили все внутренние отсеки (и интерфейсы SATA) в нашем небольшом корпусе, поэтому наша ОС отправилась на симпатичный гибридный диск WD Black2 USB3. И это очень



► Наш массив вполне справился с тестом произвольной записи AIO, а с IOZone вообще замечательно.

неплохо, если только вы случайно не выдернете этот диск во время работы машины. Для простого устройства NAS не нужна полноценная среда рабочего стола, поэтому мы начнем с установки простого Arch Linux. Если вы захотите добавить функции медиа-центра, Arch ничуть не будет против. Об установке Arch на USB-диск можно прочитать на <http://bit.ly/ArchOnAUSBKey>; остальная часть нашего руководства будет относиться и к другим дистрибутивам тоже, и мы предполагаем, что у вас имеется базовая установка с рабочим интернет-соединением. Неплохо было бы настроить на вашем компьютере демон SSH (на случай, если что-то пойдет не так) и статический IP-адрес. Эти шаги хорошо документированы, и мы будем

считать, что вы с ними справились. Итак, отключите монитор и клавиатуру и продолжите создавать устройство удаленно.

Сначала нужно разбить свои диски на разделы. Если эти диски более 2,2 ТБ, следует использовать таблицу разделов GPT. Даже если они не больше, все равно можете сделать именно так. Здесь вам поможет программа *gdisk*, это часть пакета *gptfdisk* в Arch:

```
# gdisk /dev/sda
```

Создайте новый раздел, введя *n*, затем опять нажмите на *Enter*, приняв, что это — первый раздел, и нажмите на *Enter* еще раз, чтобы принять сектор загрузки по умолчанию [2048]. Неплохо бы в конце каждого диска оставить не менее 100 МБ »

Выбор компонентов

Вам не стоит особо переживать по поводу другого оборудования, кроме дисков. Компьютеру незачем быть мощным, ему не нужна затейливая графика, и если вы не собираетесь использовать ZFS (см. стр. 46), 4 Гб ОЗУ будет более чем достаточно. Популярное решение — HP Microservers, но у них не самые элегантные корпуса. И ведь это так приятно — создавать что-то самим. Возможно, у вас завалялся где-нибудь корпус micro ATX, а если нет, вы вполне сможете собрать весьма симпатичный mini ITX, не вводя в особые расходы.

Если ваша машина будет работать 7/24 в вашей гостиной, то вам, вероятно, понадобятся тихие компоненты. И позаботьтесь, чтобы воздух нормально циркулировал вокруг дисков, не давая им перегреваться.

Мы-то, однако, выбрали AMD Kabini 5350 APU (четырёхъядерный, 2,05 ГГц, видеокарта R3). Серия Kabini, предназначенная для быстрорастущих малобюджетных рынков, была запущена в апреле, и отличается скромной рассеиваемой мощностью 25 Вт, так что перегрев не должен быть проблемой.

Контроллер на чипе имеет встроенную поддержку только двух дисков SATA, однако карты с 2 портами PCIeExpress довольно дешевые. У вас должна быть такая, которая поддерживает переключение на основе FIS (т.е. избегайте всего на основе чипа ASM1061). Если вы предпочитаете крутые чипы, то по качеству и цене вам подойдет Celeron за £ 190. Есть немало материнских плат Mini-ITX со встроенным процессором. Как и платы AM1, некоторые допускают питание от стандартного 19-В блока для ноутбука.

свободными, поскольку диски, заявленные на одинаковую емкость, часто на пару цилиндров различаются. Здесь вы можете либо прибегнуть к вычислениям, чтобы выяснить, каким именно сектором закончить раздел (умножьте размер вашего диска в терабайтах на 2 в степени 40, 100 раз вычтите 2 в степени 20, поделите на 512 — каждый сектор, вероятно, 512 байт — добавьте 2048, бум...), либо просто использовать, например, [b] +5999.9G [/b] для, допустим, 100 мегов, которых не хватает до 6 ТБ. Разделам RAID надо присваивать специальный тип, FD00, хотя Linux вообще-то перестал обращать на это внимание. Запишите новую таблицу разделов на диск, введя w. Повторите эти действия для всех дисков, которые хотите включить в свой массив.

Настройка массива

Самая интересная и самая длительная по времени часть — настройка массива. Главные сложности вы обойдете на уровне абстракции *mdadm*, однако позаботьтесь о правильности параметров: указанные вами разделы будут безвозвратно очищены от данных.

Например, наш массив RAID6 ожил после следующего заклинания:

```
# mdadm --create --verbose --level=6
--metadata=1.2 --chunk=256 --raid-devices=4 /
dev/md0 /dev/sda1 /dev/sdb1 /dev/sdc1 /dev/sdd1
```

Команда будет долго-долго работать в фоновом режиме (на создание нашего массива ушло 24 часа), и вы можете отслеживать прогресс по файлу состояния:

```
# cat /proc/mdstat
```

В режиме ограниченной функциональности можно начать работу со своим массивом немедленно, но раз терпение — это добродетель, не лучше ли пока почитать книгу или выпить пару чашек чая? Или погрузитесь в раздумья, а не выбрать ли за объем порции данных [chunk] (то есть тех частей, по каким ваши данные распределяются по дискам) 256K. По умолчанию это 512K, но оптимальный объем зависит от вашего оборудования и его применения. Для более крупных файлов рекомендуются более мелкие порции, чтобы раскидать данные по большему числу дисков, но при малом количестве дисков эта логика не работает. Для более мелких файлов стоит

Conveyance self-test routine
recommended polling time: (5) minutes.
SCT capabilities: (0x303d) SCT Status supported.
SCT Error Recovery Control supported.
SCT Feature Control supported.
SCT Data Table supported.

SMART Attributes Data Structure revision number: 16
Vendor Specific SMART Attributes with Thresholds:

ID#	ATTRIBUTE NAME	FLAG	VALUE	WORST	THRESH	TYPE	UPDATED	WHEN_FAILED	RAW_VALUE
1	Raw Read Error Rate	0x002f	200	200	051	Pre-fail	Always	-	0
3	Spin Up Time	0x0027	206	203	021	Pre-fail	Always	-	8691
4	Start/Stop Count	0x0032	100	100	000	Old_age	Always	-	15
5	Reallocated Sector Ct	0x0033	200	200	140	Pre-fail	Always	-	0
7	Seek Error Rate	0x002e	200	200	000	Old_age	Always	-	0
9	Power-On Hours	0x0032	100	100	000	Old_age	Always	-	48
10	Spin Retry Count	0x0032	100	253	000	Old_age	Always	-	0
11	Calibration Retry Count	0x0032	100	253	000	Old_age	Always	-	0
12	Power Cycle Count	0x0032	100	100	000	Old_age	Always	-	15
192	Power-Off Retract Count	0x0032	200	200	000	Old_age	Always	-	5
193	Load Cycle Count	0x0032	200	200	000	Old_age	Always	-	19
194	Temperature Celsius	0x0022	123	110	000	Old_age	Always	-	29
196	Reallocated Event Count	0x0032	200	200	000	Old_age	Always	-	0
197	Current Pending Sector	0x0032	200	200	000	Old_age	Always	-	0
198	Offline Uncorrectable	0x0030	100	253	000	Old_age	Offline	-	0
199	UDMA_CRC_Error_Count	0x0032	200	200	000	Old_age	Always	-	0
200	Multi_Zone_Error_Rate	0x0008	100	253	000	Old_age	Offline	-	0

SMART Error Log Version: 1
No Errors Logged

► *Smartmontools* могут учесть данные SMART вашего диска и предсказать проблемы, грозящие оборудованию.

использовать порции покрупнее, но уж не на порядок больше объема файлов, с которыми вы работаете. Если вы намерены отыскать оптимальное соотношение, придется потратить несколько часов на сравнительные тесты вашей системы. Помните: даже если доступ к NAS у вас через Gigabit Ethernet, узким местом, вероятнее всего, будет

сеть, так что в этом смысле особо тонко настраивать параметры RAID означает заниматься ерундой. Однако заданный параметр будет важен при инициализации нашей файловой системы.

Надо сообщить *mdadm* о своем массиве, чтобы обеспечить легкий доступ к нему после загрузки. Сделайте это, запустив

```
# mdadm --detail --scan >> /etc/mdadm.conf
```

что добавит к файлу настройки *mdadm* примерно такую строку:

```
ARRAY /dev/md0 metadata=1.2 name=wdarray:0
UUID=35f2b7a0:91b86477:b ff71c2f:abc04162
```

Теперь с узлом устройства */dev/md0* можно обращаться как с любым другим разделом, пусть даже в нашем случае это увесистые 12 ТБ. Давайте же отформатируем его, готовя к массивному притоку данных. Мы применим ext4, что, вероятно, покажется консервативным выбором, но это надежная и современная файловая система. Более экзотические варианты хорошо масштабируются на десятки дисков и могут даже управлять вашим массивом независимо от *mdadm*, но ZFS требует много памяти (настоятельно рекомендуется дорогая память ECC), а Btrfs может привести к всплескам нагрузки на CPU. Ext4 имеет пару специальных опций RAID — **stride** и **stripe-width**, и важно правильно их понимать, чтобы выровнять



► С задачами монтирования SMB управится Qt-порт *PcManFM*. Он даже не будет приставать к вам с паролем для гостевых разделов.

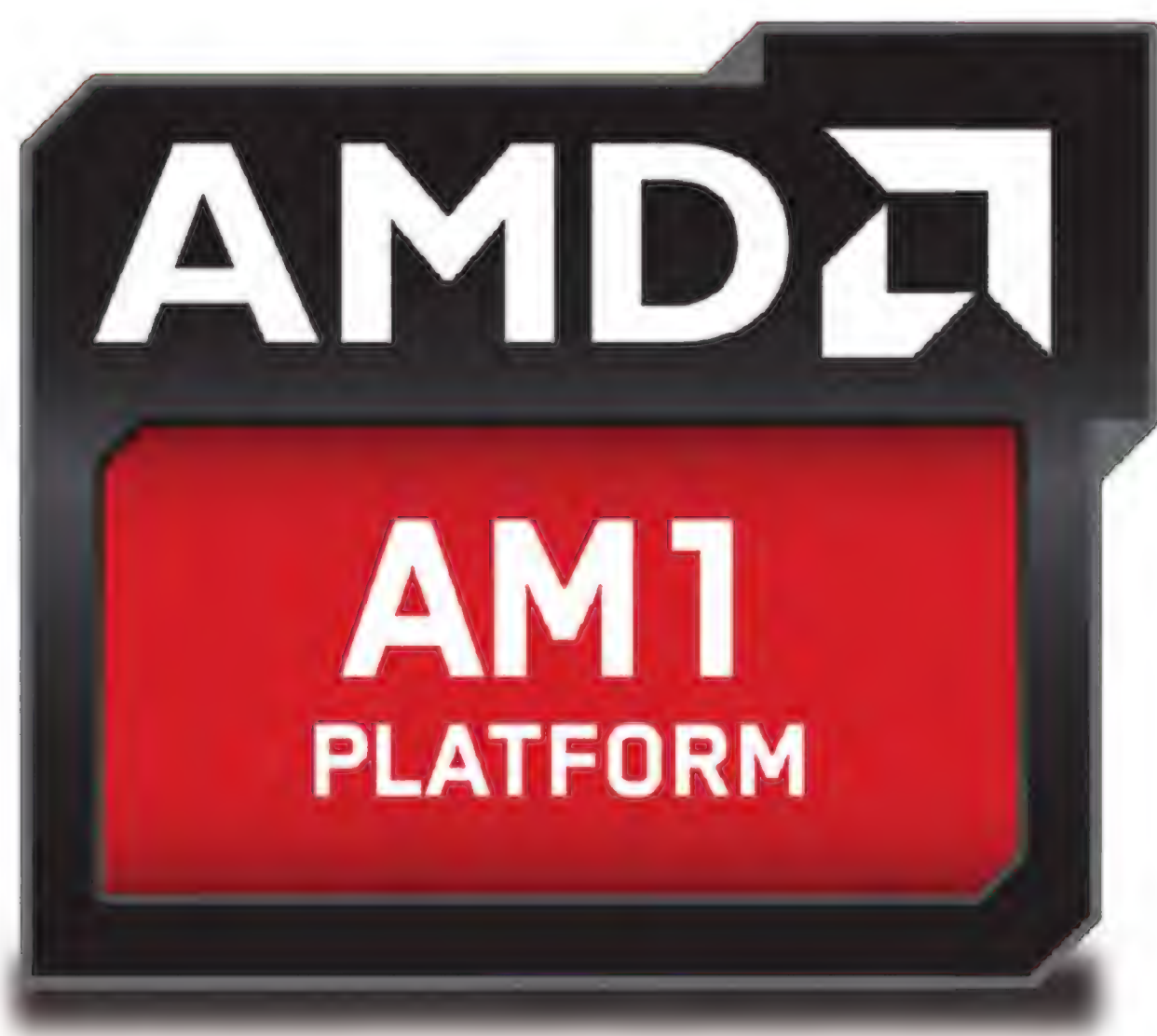
Внимание: RAID

Это предупреждение красуется в любой статье о RAID, но мы все равно добавим его и в нашу: RAID — это не резервное копирование! Это лишь первая линия обороны, и она не защитит вас от случайного удаления информации. (Как и от пожара, электромагнитных импульсов и черной магии.) Если ваши данные действительно важны, сделайте их резервную копию на внешнем носителе. Наше руководство показывает, как настроить программный RAID, и стоит развенчать ряд мифов на эту тему.

Есть специальные контроллеры RAID, хотя и по более высокой цене. По большей части особой

нужды в них нет — конечно, расчет битов четности малость нагружает процессор, но на современном железе это пренебрежимо мало. Кроме того, контроллеры оборудования обычно используют проприетарные структуры диска, и если ваш контроллер откажет, придется заменить его идентичным, чтобы добраться до своих дисков. Доступ к программному массиву RAID можно получить с помощью любой ОС Linux через команду *mdadm*. Аппаратные контроллеры RAID бывают придирчивы к совместимости дисков SATA; а при программном RAID, если ОС видит диск, то управится и с RAID.

И, наконец, ваша материнская плата может заявить, что она поддерживает различные конфигурации RAID. Это именуется FakeRAID [ложный RAID], или иногда host RAID. Несмотря на слегка презрительное название (встроенный контроллер перекладывает все вычисления RAID на CPU), это все же стабильная структура (хотя обычно поддерживает только RAID 0, 1 и 10), позволяющая порционно распределить ваш загрузочный диск и иногда даже восстановить ваш массив из BIOS. Однако восстановление может потребовать применения программ Windows. Увы, но это так.



➤ **AMD APU Athlon 5350** — дешевый и бойкий, и более чем мощный для скромного устройства NAS.

порции и блоки файловой системы RAID. Шаг [Stride] — это количество блоков файловой системы в каждой порции, вычисляемое как размер порции/блок. Ext4 по умолчанию использует блоки 4K (хотя можно указать величину поменьше через опцию **-b**), так что каждая из наших порций объемом 256K содержит 64 блока. Ширина шага — это данная цифра, помноженная на количество дисков с данными. В нашем массиве из четырех дисков каждая порция распределена по двум дискам, а два других отведены под четность порции; таким образом, наша ширина шага составляет 128 блоков. Используем мы RAID5, диском четности был бы только один диск, но мы его не используем, поэтому отформатировали массив следующим образом:

```
# mkfs.ext4 -v -L wdarray -m 0.5 -E
stride=64,stripe-width=128 /dev/md0
```

Опция **-m** настраивает долю раздела в процентах, отведенную для суперпользователя. По умолчанию это 5%, что для более крупных томов многовато.

Samba: потанцуем?

Нам нужно добавить как минимум одного пользователя — дадим ему имя **lxraid** — и изменить разрешения на **/mnt/raid**, выдав ему доступ к нашим данным:

```
# groupadd raidusers
# useradd -m -G raidusers -s /bin/bash lxraid
# chown root:raidusers /mnt/raid
# chmod 775 /mnt/raid
```

Теперь можно устанавливать все необходимые пакеты:

```
# pacman -S samba
```

Помимо наличия учетной записи в системе, пользователи *Samba* должны иметь запись в файле **smbpasswd**. Это делается с помощью

```
# smbpasswd -a lxraid
```

Затем читаем в редакторе файл **/etc/samba/smb.conf**: здесь есть пара вещей, которые вы, возможно, решите включить или выключить. Сначала раскомментируйте и отредактируйте строку

➤ **24 ТБ отличного хранилища Western Digital Red. Емко.**



hosts allow, чтобы ограничить доступ к *Samba* локальной сетью (например, 192.168. или 192.168.0.) и внутренним интерфейсом. Ближе к концу файла вы найдете раздел Share Definitions. Добавьте следующий блок, с данными для нашего массива:

```
[raid]
comment = LXF RAID
path = /mnt/raid
public = no
valid users = lxraid
writable = yes
```

Позднее вы сможете добавить и других, включить их в группу **raidusers** и настроить для них раз-

```
;writable = yes
```

Альтернатива, разрешающая доступ на запись только пользователям из группы **raidusers** —

```
# chown lxraid:raidusers
# chmod 775 /mnt/raid/public
```

плюс добавление в определение **[public]** строки **writelist = @raidusers**

Теперь мы можем запустить сервисы *Samba* и проверить наш сервер:

```
# systemctl start {smbd,nmbd}
```

У вас должен быть доступ к разделам *Samba* из любого места вашей сети. Сервис *nmbd* позволит найти ваши разделы *Samba* через URI **\\host-**

name (Windows) или **smb://hostname** (Mac/Linux). Однако при этом возможны капризы, и удачнее будет обойтись IP-адресами. *Nautilus* и *Dolphin* позволят вам просматривать такие

ресурсы из раздела Network. Машинам, которым нужно видеть разделы SMB, следует как минимум установить пакет *smbclient*, чтобы просматривать и монтировать сетевые разделы. Если вам не удалось получить доступ к своим разделам, тогда удачи вам в поиске проблем — неплохим началом будет команда **testparm**, она проверит ваш **smb.conf** на предмет аномалий. Затем можете настроить эти сервисы на автоматический запуск, заменив **start** на **enable**.

Если вы хотите автоматически монтировать свой сервер с другой машины в сети, можно добавить в **/etc/fstab** нечто вроде **//192.168.133.225/raid /mnt/raid cifs username=lxraid, password=password 0 0**

«Главные сложности вы обойдете на уровне абстракции mdadm.»

решения. Желая настроить публичную область, где не требуется аутентификация, сначала создайте директорию:

```
# mkdir /mnt/raid/public
```

По умолчанию непривилегированному пользователю **nobody** приписывается гостевой доступ, а если требуется доступ на запись **[write]**, надо сделать в этой директории **chmod 777** и раскомментировать последнюю строку в следующем определении:

```
[public]
comment = Guest area
path = /mnt/raid/public
public = yes
read only = yes
```


➤ Материнские платы AM1 имеют ограничения по части выбора и функций, но зато они довольно компактны.



Подобное хранение паролей не является безопасным; используйте его только на тех компьютерах, где под-считывать некому. Кроме того, если, в силу любых причин сервис *Samba* не работает — скажем, компьютер выключен — то на машине с подобной записью в *fstab* запуск будет замедленным.

Открываемся миру

Иногда желательно сделать свой NAS доступным через Интернет. К этому не стоит относиться легко, и это отнюдь не стоит делать с помощью *Samba*. Одна из возможностей — перенаправить порт 22 на вашем роутере на ваш NAS и соединяться через SFTP. Вы должны защитить SSH-сервер — в частности, неплохо использовать строку `PermitRootLogin without-password`

Это ограничит доступ root только публичным ключом, так что вам придется сгенерировать этот ключ через `ssh-keygen` и добавить его в файл `/root/authorized_keys`.

Можно также отключить и логины root, установив вышеупомянутую опцию в `no`, однако тогда придется выполнять все задачи администрирования локально, то есть подключить к устройству NAS клавиатуру и монитор.

В качестве альтернативы, воспользуйтесь нашим руководством по *ownCloud* [Учебники LXF190, стр. 64] и добавьте раздел *Samba* во внешнем хранилище.

Поскольку ваш внешний IP-адрес, вероятно, изменится, вам нужно настроить динамический DNS с помощью такого сервиса, как DuckDNS, dyndns или no-ip. Эти сервисы позволяют запускать на вашей машине скрипт или клиентскую программу, которая будет обновлять схему DNS

«Иногда желательно сделать свой NAS доступным через Интернет.»

для вашего IP-адреса. Настроив процедуру *cron*, чтобы она периодически запускала ее на вашей машине, вы обеспечите постоянный доступ к своей машине через постоянное доменное имя. Подписаться несложно на любой из этих сервисов, но DuckDNS отличается тем, что позволяет вам всё обновлять посредством простого скрипта, в соответствии с философией Arch — KISS [Keep It Simple, Stupid — «будь проще, дурень»]. Для его

настройки следуйте инструкциям, приведенным во врезке внизу.

По большей части ваш RAID сам о себе позаботится. Но стоит приглядывать за `/proc/mdstat` — запись типа [UUUU] говорит о том, что все работает нормально, а если какой-то диск засбоит, в этом ряду появится буква F. Кроме того, можно добыть информацию из

```
# mdadm --detail /dev/md0
```

Любителям более практического подхода понравится регулярное «патрулирование памяти» своего массива. (В конце концов, в этом нет ничего плохого.) Это проверка на наличие несоответствий

между данными и блоками четности, а затем автоматический их ремонт. Запуск данной процедуры —

```
# echo check > /sys/block/md0/md/sync_action
```

Это действие потребует времени — файл `mdstat` будет отслеживать состояние, но если вы в любой момент решите его отменить, скомандуйте

```
# echo idle > /sys/block/md0/md/sync_action
```

С устройством NAS можно сделать еще много интересного, но на данном этапе мы исчерпали отведенный нам лимит. Почему бы вам не написать нам и не рассказать о своих приключениях, связанных с NAS? LXF

Настройка DuckDNS

На сервис DuckDNS можно подписаться на www.duckdns.org через Twitter, Facebook, Reddit или Google+. А если вы — явление нестандартное, придется вам решить, какое из этих четырех зол для вас наименьшее, и создать специальную учетную запись. Перейдите на пользователя lxfrad и создайте наш скрипт обновления DuckDNS:

```
# su lxfrad
$ mkdir ~/duckdns
$ cd duckdns
$ nano duck.sh
```

Введите следующее, соответственно заменив домены и токены.

```
echo url="https://www.duckdns.org/
update?domains=your_domain&token=your_
token&ip=" | curl -k -o ~/duckdns/duck.log -K -
```

Затем сделайте его исполняемым и запустите, чтобы протестировать:

```
$ chmod 700 duck.sh
$ ./duck.sh
```

Вы должны получить некий результат от `curl`, и, надо надеяться, файл `duck.log` теперь содержит

многообещающий текст **OK**. Чтобы запускать сервис автоматически, нужно установить демон *cron* — давайте используем простой *cronie*:

```
# pacman -S cronie
```

Следующая команда (от имени пользователя lxfrad) откроет пустую *crontab* (в редакторе *nano*, чтобы не вызвать неизбежную ревность *vi*):

```
$ EDITOR=nano crontab -e
```

Добавьте следующую строку, чтобы запускать `duck.sh` каждые пять минут:

```
*/*5 * * * * ~/duckdns/duck.sh >/dev/null 2>&1
```


ПОДПИШИСЬ НА LINUX FORMAT!

Оформи подписку на печатную версию журнала Linux Format на 2015 год и получи в подарок диск с архивом номеров 2005–2014, а также подписку на электронную версию в формате PDF



SHOP.LINUXFORMAT.RU

Варианты доставки

- » Почтой по России простой или заказной бандеролью
- » Курьерской службой СПСР или IML
- » В виде PDF-файлов для подписчиков электронной версии

Редакция

- » Санкт-Петербург: пр. Медиков, 5, корп. 7, тел. (812) 309-0686
- » Москва: тел. (499) 271-4954

Файловые системы: Новое поколение

Джонни Бидвелл вооружит вас знаниями о ZFS и btrfs, двух наиболее обсуждаемых файловых системах.



Как создать отличный NAS с 24 ТБ дисков в массиве RAID 6 с ext4, ищите на стр. 44. А в данной статье мы покажем, как настроить альтернативную файловую систему.

Хотя ext4 прекрасно подходит для дисков объемом до 100 ТБ, даже ее главный разработчик Тед Цо [Ted Tso] признавал, что эта файловая система — просто временная замена ext3, которая должна была устранить ее недостатки, сохранив обратную совместимость. Ext4 впервые появилась в ядре в 2008 году; до нее самой впечатляющей файловой системой была ReiserFS, обладавшая рядом по-настоящему инновационных возможностей, в том числе комбинированными древовидными структурами B+ для метаданных файлов и списков каталогов (аналогично btrfs). Впрочем, интерес к этой системе немного ослабел, когда ее создателя Ханса Райзера [Hans Reiser] осудили за убийство жены. Разработка Reiser4 в его отсутствие продолжилась, но пока у разработчиков нет планов по ее включению в ядро.

Но теперь у нас появилось новое поколение файловых систем, гарантирующих наилучшую

целостность данных и невероятную масштабируемость. Эти файловые системы ломают и несколько старых правил: по традиционным технологиям, слой RAID (в виде аппаратного контроллера или программного менеджера, такого как *mdadm*) должен быть независим от файловой системы, и эти двое должны быть в блаженном неведении друг относительно друга. Но интегрировав их, мы сможем улучшить определение и исправление ошибок, пусть и ценой причитаний традиционалистов о «вопиющих нарушениях иерархии».

«Интерес к ReiserFS ослабел, когда ее создателя осудили за убийство жены.»

(Сравнительно) новые файловые системы — btrfs (файловая система B-дерева: произносится как «баттер-ФС» [butter-FS] или «беттер-ФС» [better-FS]), разработанная совместно Oracle, Red Hat, Intel, SUSE и многими другими компаниями, и ZFS, разработанная Sun Microsystems до ее поглощения Oracle. Код ZFS был изначально выпущен в 2005 году как часть OpenSolaris,

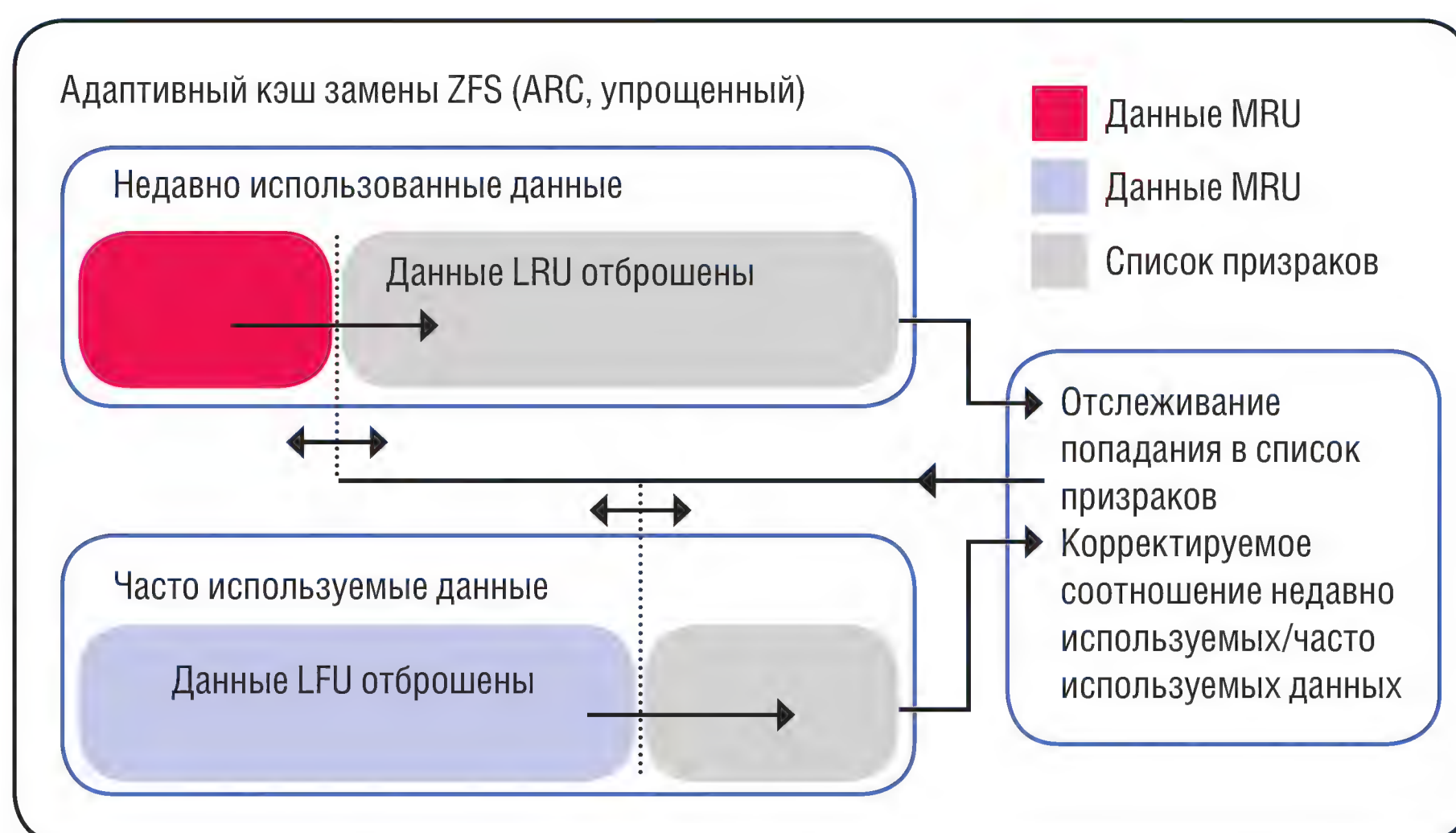
но с 2010 года эта разработка остановилась, и теперь ZFS в Solaris разрабатывается как закрытый проект. Разработка версии с открытым исходным кодом продолжается отдельно, но поскольку у ZFS лицензия CDDL, несовместимая с GPL, ее поддержку невозможно включить в ядро Linux напрямую. Однако поддержка посредством стороннего модуля тем не менее кошерна, и именно этим занимается проект ZFS on Linux (<http://zfsonlinux.org>). Этот проект в основном финансируется Ливерморской национальной лабораторией, у которой достаточно большие требования к размеру хранилищ, поэтому ZFS поддерживает размеры файлов до 16 эксабайт (2^{24} ТБ) и размеры томов до 256 зетабайт (2^{38} ТБ).

Будучи по отношению к ядру сторонним модулем, ZFS чувствительна к обновлениям ядра. В дистрибутивах на базе Debian, Fedora, CentOS и других об этом позаботятся пакеты типа DKMS, но в других дистрибутивах придется перекомпилировать модуль при каждом обновлении ядра. Если этого не сделать, и ваша корневая файловая система — ZFS, то могут возникнуть проблемы. Пользователям Ubuntu нужно добавить PPA *zfs-native/stable* и установить пакет *ubuntu-zfs*.

На странице ZFS on Linux можно отыскать пакеты и информацию для всех пользователей.

Начнем с общей почвы этих файловых систем. Одно из их поразительных свойств в том, что ни одна из этих файловых систем не требует разбивать диски на разделы. В ZFS можно настроить наборы данных с однодисковым `zpool`, который предоставляет лучшую изоляцию по сравнению с каталогами, а также может включить квоты и другие средства контроля. Обычные разделы в `btrfs` также можно представить в виде подтомов. В обоих случаях результат гораздо более гибок — «неоразделы» гораздо проще сочетать или менять их размер, так как это структуры чисто логические. ZFS активно противится использованию себя напрямую на разделах, а `btrfs` по большому счету все равно.

Обе файловые системы включают менеджер логических томов, позволяющий файловой системе охватывать несколько дисков и содержащий подструктуры с различными именами. Обе файловые системы также имеют собственные реализации RAID, хотя, и это сбивает с толку, их уровни RAID не совпадают с традиционными: в ZFS есть три уровня четности RAID под названиями RAID-Z1, -Z2 и -Z3. По сути это то же, что RAID5, RAID6 и теоретический RAID7, т.е. для четности в них используется 1, 2 и 3 диска, и поэтому они могут справиться с выходом из строя несколь-



➤ **Кэширование в ZFS: два списка, для недавно использованных и часто используемых данных, разделяют общую память. Только что использованные данные (MRU) сохраняются слева и попадают в «список призраков», если к ним не обращаются. Память соразмерно распределяется в зависимости от того, как часто происходит обращение к данным в «списке призраков».**

и полезная емкость (из предыдущего примера) составит 1,5 ТБ ценой чуть меньшей избыточности. В `btrfs` RAID1 также можно использовать несколько дисков разного размера, но при этом в зависимости от их сочетания может быть какое-то количество недоступного для использования места

(так как доступно менее половины суммарного хранилища). `Btrfs` также позволяет указать различные уровни RAID для данных и метаданных; ZFS реали-

зует зеркалирование почти как RAID1, но не называет его этим именем.

Вообще, зеркалирование в обеих файловых системах устроено сложнее по сравнению с традиционным RAID, так как ошибки определяются и устраняются автоматически. Если в традиционном зеркале RAID1 блок поврежден (но остался читаемым) на одном диске и уцелел на другом, `mdadm` не может понять, на котором диске данные правильные; в половине случаев будет читаться хороший блок, а в половине случаев — плохой.

Такие ошибки-«тихари» — сущий бич: гораздо проще понять, когда диск перестал отвечать, с чем и должен бороться RAID. В ZFS хранятся свертки SHA-256 для каждого блока, а в `btrfs` используются контрольные суммы CRC32C — и для данных, и для метаданных. Обе файловые системы определяют и молча устраняют несоответствие при определении неисправного блока. Можно и нужно периодически выполнять очистку одного из томов следующего уровня. Это проверка по ходу (размонтировать пулы не нужно), которая выполняется в фоне и определяет и исправляет все ошибки.

Вся эта КоЗа (копирование при записи — CoW, Copy-on-Writing) может привести к излишней фрагментации, проявляющейся в заметной пробуксовке диска и пиках нагрузки процессора, но есть защитные средства, способные снизить влияние этих факторов. В ZFS пространство выделяется большими блоками по 128 КБ, а в `btrfs` используются В-деревья. Идея в основе обоих подходов одна и та же: заранее выделить важные области диска для новых данных. В отличие от `btrfs`, в ZFS нет дефрагментации, что способно привести к заметному падению производительности

«Поразительно: ни одна из этих ФС не требует разбивать диски на разделы.»

ких дисков. RAID5 и 6 поддерживаются в `btrfs`, но использовать их в реальных системах не стоит, так как эта часть кода значительно менее зрелая по сравнению с остальной. Обе файловые системы имеют стабильную поддержку RAID 0, 1 и 10, но эти уровни опять же интерпретируются немного по-другому. Например, традиционный массив RAID 1 на трех дисках размером 1 ТБ отзеркалировал бы данные дважды, и полезная емкость составила бы 1 ТБ. Но в `btrfs` RAID 1 означает, что каждый блок один раз зеркалируется на другом диске,

Краткая история файловых систем

Сначала данные записывались на перфокартах и магнитной ленте. Понятия файла не существовало: все данные хранились в одном потоке. В этом потоке можно было указывать на различные адреса (например, перематывать пленку по счетчику, чтобы найти место записи), но по сути это был один аморфный массив данных.

В середине восьмидесятых появились файловые системы с одним каталогом, или плоские. В них на устройстве можно было размещать отдельные файлы, но не подкаталоги. Их появление совпало с растущей популярностью дискет, допускающих произвольный доступ к данным (можно было

считывать/записывать данные в любое место на диске). В ранних файловых менеджерах Mac на основе плоской файловой системы создавалась иерархическая структура каталогов, но при этом файлы все равно обязаны были иметь уникальные имена.

К концу 1980-х в связи с развитием технологии хранения данных и возрастающей сложностью операционных систем потребовались файловые системы с подходящими каталогами. Такие системы появились начиная с IBM PC-DOS версии 2, и первая из них называлась FAT16. Она допускала имена файлов по стандарту 8.3 и тома размером

до 2 ГБ. В Windows 95 появились длинные имена файлов и возможность доступа к дискам свыше 8 ГБ, но пользователям Linux эти преимущества стали доступны уже с 1993 года благодаря `ext2`. Эта файловая система ознаменовала новый шаг вперед: в ней появились метаданные, такие как права доступа к файлам, и файловая система стала неразрывно связана с механизмом контроля доступа. В `ext3` и более поздних версиях NTFS ввели следующую инновацию — журналирование, позволившее легко проверять целостность файловых систем и быстро восстанавливать файловую систему из-за сбоя ОС или отключения питания.

при заполнении пулов неправильными файлами, но с домашними хранилищами это маловероятно, особенно если общий размер хранимых файлов не превышает 60% емкости хранилища. Если вы знаете, что у вас есть файл, не совместимый с CoW, например, большой файл, подвергающийся множеству малых случайных операций записи (пусть этот файл называется **ruminophobe**), можно задать расширенный атрибут **C**, который изменит традиционную перезапись для этого файла:

```
$ chattr +C ruminophobe
```

Этот флаг действителен и для btrfs, и для ZFS, а по сути — для любой файловой системы с поддержкой CoW. Его можно применить и к каталогам, но это повлияет только на файлы, добавленные в каталог после применения флага. Также можно воспользоваться атрибутом **c** для включения сжатия. Его также можно задать на уровне тома с помощью параметра монтирования **compress**. Обе файловые системы позволяют выполнять сжатие в *zlib*, которое не стоит включать, если вы не согласны на серьезное падение производительности. Btrfs предлагает *LZO*, которая сильно не навредит, даже если у вас много уже сжатых

данных. ZFS предлагает алгоритмы LZJB и LZ4, а также «родной» ZLE (схема с нулевой длиной кодировки — Zero Length Encoding) и возможность указывать уровни сжатия *zlib*.

Обратите внимание, что хотя и btrfs, и ZFS — файловые системы следующего поколения и их возможности во многом пересекаются, это различные создания, и у каждой из них свои преимущества и недостатки, капризы и странности.

Поговорим о ZFS, детка

Фундаментальная единица хранения данных в ZFS называется **vdev**. Это может быть диск, раздел (не рекомендуется), файл или даже набор таких **vdev**, например, зеркало или RAID-Z из нескольких дисков. Один или несколько **vdev** образуют пул хранения данных, или **zpool**. Устройства в **zpool** можно добавлять по требованию, и при добавлении пула все файловые системы (точнее, «наборы данных») получают новое пространство мгновенно. На рисунке ниже показан пример в ZFS, эквивалентный массиву RAID10, когда данные зеркалируются между двумя дисками и затем разделяются по еще одной паре зеркалируемых дисков. Каждая

зеркалируемая пара также представляет собой **vdev**, а вместе они образуют наш пул.

Пусть модуль ZFS установлен и загружен, и вы собираетесь настроить пул, раскиданный по нескольким дискам. Но нужно убедиться, что на дисках нет информации RAID, иначе ZFS встанет в тупик. Для этого рекомендуется определить идентификаторы дисков. Имена **/dev/sdX** также подойдут, но они могут меняться, поэтому воспользуйтесь командой

```
# ls -l /dev/disk/by-id
```

и затем используйте соответствующие идентификаторы в следующей команде, которая создаст пул под названием **tank**:

```
# zpool create -m <mountpoint> tank <ids>
```

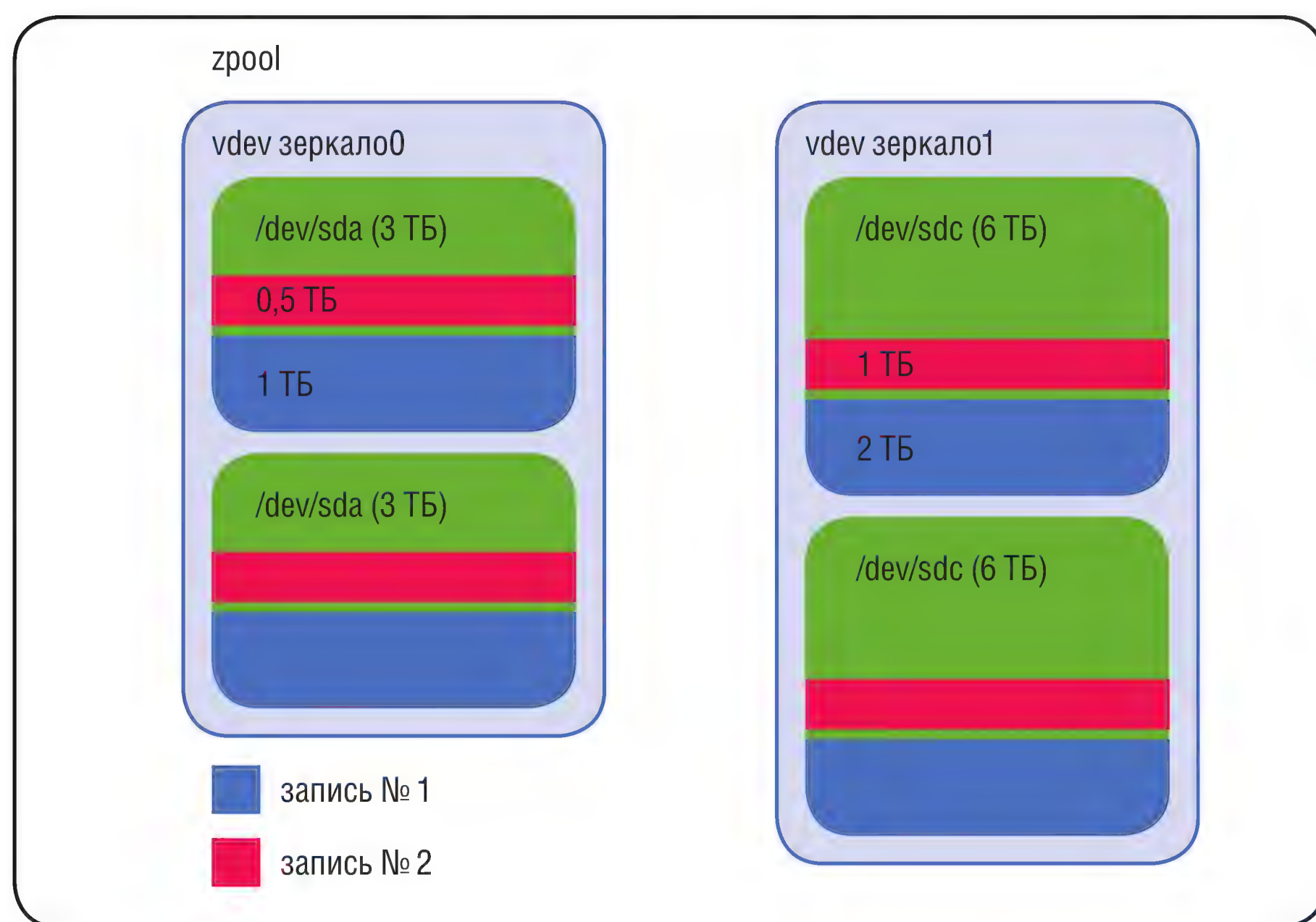
Если у вас новые диски (выпущенные после 2010 года), то на них скорее всего будут сектора по 4 КБ в противоположность старым дискам с размером сектора 512 байт. ZFS умеет работать с любым размером сектора, но некоторые более новые диски эмулируют поведение старых, чтобы ими можно было пользоваться в Windows 95, и это сбивает ZFS с толку. Чтобы принудительно выровнять пул по секторам новых дисков, добавьте к приведенной выше команде **-o ashift=12**. В ней также не обязательно указывать точку монтирования: в нашем случае, по умолчанию будет взята **/tank**. Для настройки зеркал используется ключевое слово **mirror**, поэтому пул в стиле RAID10 на схеме внизу (где у нас не хватило места на идентификаторы дисков, но вы-то должны их использовать) можно задать командой

```
# zpool create -o ashift=12 mirror tank mirror /dev/sda /dev/sdb mirror /dev/sdc /dev/sdd
```

Если вы хотите вместо этого настроить RAID-Z1, воспользуйтесь ключевым словом **raidz1**, заменив **1** на **2** или **3**, если нужна двойная или тройная четность. После создания можно проверить состояние пула командой

```
# zpool status -v tank
```

Теперь добавляйте в **zpool** файлы и каталоги, как и в любой другой монтируемой файловой системе. Но также можно добавить файловые системы (другие, особого ZFS-типа), тома (**zvol**), снимки (**snapshot**) и клоны (**clone**). Четыре этих понятия совместно называются наборами данных, и в ZFS с ними можно выполнять массу различных действий. Файловая система в пуле ZFS чем-то напоминает раздел диска, но более проста



➤ ZFS распределит данные умно, с учетом доступного пространства: после записи 3 ТБ и затем 1,5 ТБ все диски будут наполовину заняты (или полупусты, если вы пессимист).

Вот тебе CoW, приятель

Даже если в вашей файловой системе (ФС) нового поколения нет избыточности, она будет значительно надежнее своих предшественников. Это возможно благодаря технологии, называемой «копированием при записи (Copy-on-Write, CoW)»: новая версия файла не перезаписывает старую, а записывается в другое место на диске. После этого, и лишь после этого, новое расположение файла прописывается в его метаданных, и ранее занятое пространство освобождается. Это означает, что при падении системы или при отключении питания во время записи

вместо поврежденного файла у вас по крайней мере будет корректная копия старого. Наряду с повышением надежности CoW позволяет легко делать снимки ФС (или, точнее, подтома). Снимки — одна из функций (если не главная функция) наших ФС нового поколения. Снимок выглядит как побайтовая копия подтома в заданный момент времени (пока можно считать, что подтом — то же, что и старый добрый «каталог» — но фактическое определение для btrfs и ZFS отличается), но при первом создании снимка он почти не занимает места на диске. Сперва

снимок просто ссылается на исходный подтом. При изменении данных на исходном подтоме надо сохранить их на снимке, но благодаря CoW исходные данные все еще на диске; снимок просто ссылается на старые данные, а ФС не помечает эти блоки как неиспользуемые, и в результате старые и новые версии файла могут жить бок о бок. Это позволяет легко делать ежедневные снимки целой ФС, если большая часть файлов в ней не меняется кардинально. Снимки можно даже дублировать на удаленные пулы через SSH.

в создании и изменении размера (изменение размера в смысле ограничения максимального размера квотой). Также можно настроить сжатие для каждой файловой системы.

Давайте создадим простую файловую систему, с именем **stuff**. Обратите внимание, что наш пул **tank** не получает спереди /, когда мы ссылаемся на него инструментами ZFS. Нам не надо, чтобы он был слишком велик, так что мы наложим на него квоту 10 ГБ и, наконец, проверим, что все прошло нормально:

```
# zfs create tank/stuff
# zfs set quota=10G tank/stuff
# zfs list
```

Zvol — странная структура: это виртуальное блочное устройство. Для обращения к zvol используется узел **/dev**, и как и любое другое блочное устройство, его можно отформатировать с какой-нибудь файловой системой. Что бы вы ни делали с zvol, на все это распространяются обычные возможности zpool, т.е. его можно зеркалировать, сжать и легко сделать снимок. Мы уже рассказали о снимках (см. врезку «Вот тебе CoW, приятель»), но в ZFS есть несколько особых моментов. Во-первых, нельзя делать снимки каталогов — только файловых систем. Поэтому давайте сделаем снимок нашей файловой системы **stuff** и подивимся, как мало места он займет:

```
# zfs snapshot tank/stuff@snapshot0
# zfs list -t all
```

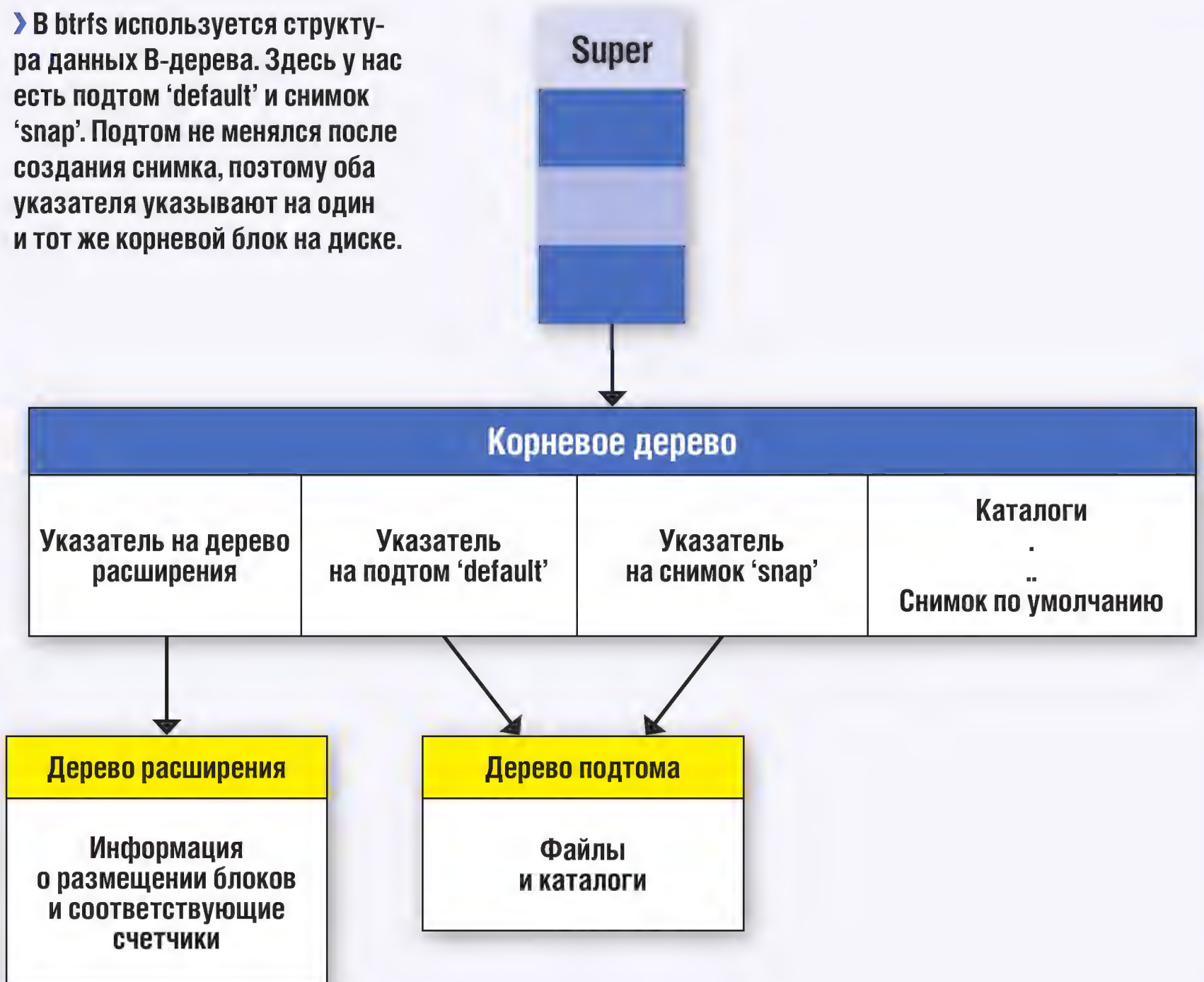
Синтаксис с собачкой напоминает цели *systemd*... но не будем отвлекаться. Свой снимок вы можете назвать пооригинальнее, чем **snapshot0** — пожалуй, стоит включить в название снимка дату или как-то еще обозначить момент его создания. Допустим, мы сделали нечто безрассудное, и набор данных **stuff** оказался испорчен. Не проблема: можно откатиться к моменту снимка **snapshot0** и попытаться не повторять этой ошибки. Команда **zfs diff** покажет даже созданные (+), измененные (M) и удаленные (-) файлы с момента создания снимка:

```
# zfs diff tank/stuff@snapshot0
M /pool/stuff
+ /pool/stuff/newfile
- /pool/stuff/oldfile
# zfs rollback tank/stuff@snapshot0
```

Снимок доступен только на чтение, но можно создать его эквивалент, доступный на запись: это последний участник квартета «наборы данных» — «клон».

Нельзя не упомянуть, что ZFS лучше всего работает при большом объеме оперативной памяти. Некоторые рекомендации доходят до 1 ГБ на 1 ТБ дискового пространства, но в зависимости от ваших целей можно обойтись и меньшим. Одна из причин такой требовательности к памяти — адаптивный кэш замен ZFS (Adaptive Replacement Cache). Это улучшение запатентованного механизма IBM ARC: благодаря учету как последнего, так и частого доступа (см. схему на стр. 51) он обеспечивает высокую частоту попадания в кэш. По умолчанию он использует до 60% доступной памяти, но этот показатель можно настроить с помощью параметра **zfs_arc**

➤ В btrfs используется структура данных B-дерева. Здесь у нас есть подтом 'default' и снимок 'snap'. Подтом не менялся после создания снимка, поэтому оба указателя указывают на один и тот же корневой блок на диске.



max, задающего предельный размер кэша в байтах. Если вы используете дедупликацию, памяти действительно должно быть много — около 5 ГБ на 1 ТБ — и мы такого не советуем. Последнее предупреждение: используйте оперативную память с ECC (кодом коррекции ошибок). Все достоинства контрольных сумм ZFS будут в лучшем случае бесполезными, а в худшем — опасными, если при их вычислении вдруг случайно затешется какой-то шальной бит. Ошибки памяти редки, но они бывают, виноваты ли в них аппаратные сбои или космическое излучение.

Согрей мне btrfs, детка

С помощью **mkfs.btrfs** можно как создать новую файловую систему btrfs, так и конвертировать существующую ext3/4. Разумеется, во время конвертации её нельзя смонтировать, а значит, для конвертации корневой файловой системы надо загрузиться с Live CD или в другой дистрибутив Linux. Затем воспользуйтесь командой **btrfs-convert**. Эта команда изменит UUID раздела, так что подправьте *fstab* в соответствии с ним. Конвертированный раздел содержит образ старой файловой системы — на случай, если что-то пойдет не так. Этот образ хранится на подтоме btrfs, во многом похожем на набор данных файловой системы ZFS.

Как и в ZFS, можно делать снимки только подтомов, а не отдельных каталогов. Но в отличие от ZFS снимок не рекурсивен, и если сам подтом содержит другой подтом, то последний в снимке станет пустым каталогом. Так как сам снимок представляет собой подтом, можно также делать снимки снимков. Корневую файловую систему разумно разместить на подтоме btrfs, особенно если вы собираетесь делать ее снимки, но это выходит за рамки нашей статьи.

Подтомы создаются командой

```
# btrfs subvolume create <subvolume-name>
```

Они появятся в корне файловой системы btrfs, но можно смонтировать их и индивидуально, указав параметр **subvol=<имя-подтома>** команды **fstab** или **mount**. Сделать их снимки можно командой

```
# btrfs subvolume snapshot <имя-подтома>
<имя-снимка>
```

Снимок можно принудительно сделать доступным только на чтение, указав параметр **-r**. Чтобы развернуть снимок, воспользуйтесь командой

```
# btrfs subvolume snapshot <имя-снимка>
<имя-подтома>
```

Если все прошло хорошо, можно удалить исходный подтом.

Файловые системы btrfs можно оптимизировать для SSD, монтируя их с ключевыми словами **discard** и **ssd**. Даже на одном диске btrfs по умолчанию будет зеркалировать метаданные — хотя это хуже, чем держать метаданные на другом диске, но иногда бывает удобным. Если дисков несколько, btrfs по умолчанию зеркалирует метаданные в RAID 1.

Для интерактивной дефрагментации данных всех файлов в btrfs послужит команда

```
# btrfs filesystem defragment -r -v /
```

Также можно воспользоваться опцией мониторинга btrfs **autodefrag**. Еще одна полезная в хозяйстве команда — **btrfs balance**. Она перезаписывает данные и метаданные, равномерно распределяя их по нескольким дискам. Это особенно удобно, если у вас почти заполнены все диски, и вы добавляете новый командой **btrfs add**.

Конечно, об обеих файловых системах можно рассказать еще много интересного. В вики Arch Linux есть отличные руководства по btrfs (<http://bit.ly/BtrfsGuide>) и ZFS (<http://bit.ly/ZFSGuide>). **LXF**

Мод Minetest

Джонни Бидвелл учит вас перекраивать на свой вкус открытую игру в жанре песочницы по мотивам *Minecraft*.

Когда в сентябре 2014 г. Microsoft приобрёл Mojang, игроки *Minecraft* обеспокоились и испугались за будущее как игры, так и сообщества (посмотрите хотя бы вот это видео — <http://bit.ly/MSMinecraft>). Нотч [Notch, он же Маркус А. Перссон — Marcus A. Persson — создатель игры, — прим. пер.] и его команда ушли, и теперь, видимо, потягивают мохито где-нибудь на Багамах, размышляя о том, куда всадить свои \$2,5 миллиарда. Хотя в Редмонде и поклялись уважать преданное сообщество, сложившееся вокруг *Minecraft*, точно предсказать, что будет с самой популярной видеоигрой за всю историю челове-

чества, не может никто. Даже если фигура нового владельца вам не претит, неплохо всё же иметь и открытую альтернативу.

И она есть. *Minetest* (<http://minetest.net>) — это открытый (LGPL 2.1) вариант *Minecraft*. Сущест-

«Нотч и его команда сейчас, видимо, потягивают мохито на Багамах.»

вующий, на самом деле, уже около 4 лет, а значит, имеющий стабильную кодовую базу, сложившееся сообщество и собственный подфорум на сайте [reddit.com](http://reddit.com/r/Minetest) (r/Minetest).

Рождение открытых версий популярных игр — явление не новое. Прекрасный пример — *PrBoom* (<http://bit.ly/PRBoomGame>), переделка движка *Doom* (а также *Doom II*, *Heretic* и *Hexen*). Вы можете использовать файлы данных (то есть таблицы распределения памяти, графику и звуки) из вашей платной копии, или использовать ресурсы сообщества, например, из проекта *Freedoom* (<https://freedoom.github.io>). Среди других подобных переделок можно назвать *ScummVM* (<http://scummvm.org>) для ранних игр LucasArts, *OpenAge* (<http://bit.ly/OpenAgeGame>) для *Age of Empires II* и *Stratagus Core* (<http://stratagus.com>) для *StarCraft* и *Warcraft II*.

Создатели других проектов решили не пачкаться с переделкой или обратным инжинирингом вариантов оригинала, а переписать всё заново. Поскольку *Minecraft* относится к жанру песочницы, логично было бы применить тот же движок: уровней как таковых там нет, а переделывать спрайты и текстуры необычайно трудно. В игре есть режимы и выживания, и творчества, и практически любой её аспект можно воссоздать на популярном скриптовом языке Lua. Ядро *Minetest* написано в основном на C++ и использует движок Irrlicht (по-немецки «блуждающий огонёк»). Благодаря этому код очень легко портировать, и есть сборки для Linux, FreeBSD, Windows и OS X. Он также менее алчен к ресурсам, чем *Minecraft* на базе Java, и потому поладит и с более старым оборудованием (разработчики рекомендуют графику не ниже Intel 945 и двухъядерный процессор). Пакеты Minetest доступны в официальных репозиториях Ubuntu, Debian (и Raspbian), Arch Linux, Gentoo, Fedora, openSUSE и, возможно, многих других. Но если вдруг, что весьма маловероятно, в вашем дистрибутиве их нет, исходники можно скачать с сайта.

Всё установив, можете запустить игру из терминала простым \$ Minetest

Возможно, появится также и значок в меню Игры. Начните с создания нового мира, придумайте ему какое-нибудь смешное имя, нажмите кнопку Создать, а затем Играть. Бегать и прыгать вы сумеете рефлексивно, но чтобы выжить, надо научиться большому. В конце концов, ночь уже близко! Разумно было бы для начала собрать дров, так что найдите дерево и порубите его с помощью левой кнопки мыши. Свыкнувшись с ролью лесоруба, вы увидите, как готовые поленья накапливаются в одном из сундуков в низу экрана. Такой способ уборки применяется ко всем типам сырья в данном игровом мире, в том числе цветам, яблокам, гравиям, песку и не только. Чтобы сложить поленья, нажмите I — откроется окно инвентаря. Перетащите их в зону мастерской, состоящую из девяти квадратов, и сможете постепенно заполнять их, начиная справа. Поленья можно будет расколоть на палки, обладающие множеством применений. Этим освоение ремесла не исчерпывается: можно, например, изготовить кирку (на рис. стр. 56), что позволит увеличить добычу угля.

Разводим огонь

Из множества навыков, которые необходимо освоить для выживания, главным является умение развести огонь. Горящие предметы, в частности, факелы, очень пригодятся в темноте. Для факелов нужны те самые палки и уголь, добываемый свежесделанной киркой. Более серьезный огонь пригодится для плавки в печи — её можно соорудить, окружив центральный квадрат в панели любым из доступных видов камней. Как и в самом *Minecraft*, есть фиксированные рецепты (как, например,



в случае с упомянутыми киркой и печью), когда ингредиенты собираются особым образом и только в зоне мастерской, и есть рецепты свободные, где местоположение не важно. Создав печь, вы должны найти для неё какое-нибудь топливо и сырьё для выплавки. Как и в реальной жизни, горят многие вещи, но не все одинаково хорошо. Щелкнув по печи правой кнопкой мыши, вы увидите таблицу горючего. Положите топливо в нижний отсек, а то, что нужно переплавить — в верхний. Если горючее подобрано правильно, то уже через несколько секунд готовый продукт появится в правом отсеке. Полное руководство по выплавке см. на вики *Minetest* (<http://bit.ly/MTSmelting>), хотя нас огорчило, что теперь в стандартной игре нет крыс, перекус которыми в жареном виде раньше прибавлял здоровья

На самом деле, при обычном запуске в *Minetest* нет мобов (мобильных объектов), что делает его скучноватым. Но не бойтесь, в режиме Простые Мобы [Simple Mobs], который вы найдёте на **LXFDVD**, есть и крысы, и овцы, и пять видов ужасных монстров.

Вам нужно создать каталог с именем **mods** и распаковать туда архив:

```
$ mkdir ~/.minetest/mods
$ cd ~/.minetest/mods
$ tar -xvzf simple_mobs.tar.gz
```

Теперь запустите игру и выберите Включить все в меню Настройка. Надпись «Мобы» должна стать зеленой. И, вероятно, когда вы откроете свой мир, там будет бегать пара овец. Можете даже создать одну прямо около себя, нажав t (появится чат-диалог, который также функционирует в качестве консоли) и набрав

```
/spawnentity mobs:sheep
```

» Это сервер Liberty Land [Земля Свободы], где пользователи могут получить участок земли, а также оставлять письма в почтовых ящиках.

API Minetest

Minetest включает три основных класса объектов:

» **Элементы** Блоки вашего мира.

» **Инструменты** Оружие, инструменты как таковые и другие орудия для изготовления вещей.

» **Сырьё** Различные предметы (например, палки, бумага и минералы).

Когда вы дойдёте до написания собственных модов, вам придётся регистрировать любые новые объекты в файле под названием **init.lua**, используя, например, метод **minetest.register_node()**. Этот файл инициализации должен быть в любом моде, внутри

каталога **~/.minetest/mods** для пользовательских модов или **/usr/share/minetest/mods** для общих.

Помимо объектов, можно также создавать рецепты ремесел. Например, добавив следующее условие в файл **init.lua** мода мобов, мы получим радужный блок Nyan Cat с помощью яблока и блока грязи... как и в реальной жизни:

```
minetest.register_craft({
    output = "default:nyancat_rainbow" 1,
    recipe = {
        {'default:dirt', 'default:apple', ""},
```

```
{ "", "",
  {"", ""},
  }
})
```

Блоки Nyan Cat существуют и сами по себе, но это большая редкость, и найти их можно только на 32 блока ниже уровня моря. Массив **recipe** соответствует расположению элементов в сетке ремесел. Можно указать третий необязательный параметр **type**, либо как **"shapeless"** для бесформенного рецепта, либо **"cooking"**, если применяется печь.



► Солнце восходит почти ровно в 5:00, освещая останки павшего игрока... которые мы разграбили наживы ради.

Заглянув в файл `~/minetest/mods/mobs/init.lua`, вы увидите другие доступные мобы: деревянных/песочных/каменных/грязевых монстров, орков и властелинов подземелий, а также менее враждебных овец и крыс. Кроме того, изучив этот файл, вы получите представление об API Lua, применяемом всеми модами. Например, вы увидите, что по щелчку правой кнопкой мыши на овце игрок получает шерсть — если овцу не остригли до того. А также узнаете, что овцы потянутся за игроком, если тот несет пшеницу. Если вы настроены на зверства, можете убивать своих овец и есть их, приготовив в печи, для повышения здоровья. Правда, чтобы добыть мятный соус, придётся написать свой собственный мод.

В режиме Simple Mobs вы заметите, что с наступлением ночи просыпаются всякие злые силы, посягающие на вашу жизнь.

«Из навыков, необходимых для выживания, главным является умение развести огонь.»

Одних убить легко, других — труднее, но киркой всё же сподручнее, чем голыми руками, или, скажем, бараньей ногой. Вероятно и то, что вы будете молиться, чтобы поскорее настал рассвет — и при везении ваши молитвы будут услышаны, если вы успеете выполнить команду

```
/time 5000
```

прежде чем погибнуть от руки какого-нибудь зловредного существа. В *Minetest* время идёт от 0 до 23999, так что восход солнца, а с ним и прекращение появлений монстров, приходится где-то на 4500. К сожалению, те, что успели народиться, продолжают охоту за вами. Если вы хотите сжульничать и вооружиться против них, минуя лихорадочный процесс создания ремесел, скомандуйте

```
/giveme default:sword_diamond
```

Если силы тьмы всё же одолеют вас, вы сможете возродиться в другом месте. Вернувшись на место своей гибели, вы найдете свои кости, порывшись в которых, вернете все свои вещи.

Почти все, с чем вы сталкиваетесь в *Minetest*, осуществляется через моды: *Minetest* без модов — это книга без слов. При включении мода мобов вы, должно быть, заметили некоторые из них. Стандартные моды игры установлены в подкаталоге **games/minetest/mods** в разделе установки. Здесь вы можете узнать, к примеру, как растут цветы, горит огонь и взрывается тротил. Мод по умолчанию содержит, среди прочего, рецепты всех ремесел (в файле **crafting.lua**), все виды оружия и инструменты (**tools.lua**), а также все доступные типы блоков (**nodes.lua**).

В меню для создания нового мира вы можете заметить выпадающий список Margen, с рядом алгоритмов для генерации карты: v6 является текущим умолчанием; v7 — экспериментальный, и без добавки параметров будет производить только каменистые ландшафты, создавать которые довольно сложно: требуется тонкий баланс спонтанности и последовательности. Основной метод называется шумом Перлина [Perlin] и служит прекрасным средством создания биомов — регионов со схожим климатом, географией и экологией. В самой свежей версии (то есть вы собрали её из исходников или используете Arch Linux) вам доступны



► Как изготавливать кирку (и оказывать влияние на людей). Для лезвия можно использовать разные материалы.

некоторые другие экспериментальные генераторы, скажем, математический, способный создавать потусторонние 3D-фракталы.

В меню Настройки есть несколько вариантов благоустройства, которые стоит попробовать, если у вас достаточно мощная машина — в том числе шейдеры, карты рельефа и волнующаяся водная поверхность. Кто-то скажет, что они лишают игру характерного духа LEGO, но нам показалось, что это красиво, да и приятно поупражнять свою видеокарту. Если вы ищете более ярких ощущений, скачивайте пакеты разнообразных пользовательских текстур с форумов *Minetest* (<http://forum.minetest.net>). Здесь вы также найдете великое множество модов, возиться с которыми — лучший способ узнать всё об API.

Minetest-морфоза

Одна из самых интересных особенностей API — возможность применять Active Block Modifiers (ABM). Они определяют изменения, способные произойти с элементами либо с течением времени, либо при контакте с другими элементами определённого типа. Например, в файле **functions.lua** описано, как при секундном контакте с водой лава превращается в булыжник:

```
minetest.register_abm({
    nodenames = {"default:lava_flow"},
    neighbors = {"group:water"},
    interval = 1.0,
    chance = 1,
    action = function(pos, node, active_object_count,
        active_object_count_wider)
        minetest.set_node(pos, {name = "default:cobblestone"})
    end,
})
```

Аргумент **chance** определяет вероятность изменения: 1 — оно точно произойдёт, тогда как более высокие значения относятся к обратной вероятности: так, 100 обозначает вероятность один-из-ста. Блок **lava_source** интереснее реагирует при остывании — помимо превращения в обсидиан, ещё и производит радующий душу звук. Чтобы выполнялись оба эти действия, они сгруппированы в функцию, которая выглядит так:

```
default.cool_lava_source = function(pos)
    minetest.set_node(pos, {name = "default:obsidian"})
    minetest.sound_play("default_cool_lava", {pos = pos, gain = 0.25})
end
```

А параметр **action** в ABM вызывает эту функцию:

```
action = function(pos, node, active_object_count,
    active_object_count_wider)
    default.cool_lava_source(pos, node, active_object_count,
    active_object_count_wider)
```



В соответствии с функциями **getBlock()** и **setBlock()**, составляющими суть часто обсуждаемой API *Minecraft*, *Minetest* имеет аналогичные функции — **minetest.get_node()** и **minetest.set_node()**. Кроме того, есть очень удобная функция, **minetest.find_node_near()**, которая поможет найти элементы заданного типа в определенном радиусе. Вы можете написать такую и сами, используя методы **get**, пару циклов и немного Пифагора, но приятно знать, что это не обязательно. Эта функция используется, например, чтобы заставить папирус расти, когда его заросли находятся над травой и на расстоянии трех блоков до источника водоснабжения:

```
minetest.register_abm({
    nodenames = {"default:papyrus"},
    neighbors = {"default:dirt", "default:dirt_with_grass"},
    interval = 50,
    chance = 20,
    action = function(pos, node)
        pos.y = pos.y - 1
        local name = minetest.get_node(pos).name
        if name == "default:dirt" or name == "default:dirt_with_grass"
        then
            if minetest.find_node_near(pos, 3, {"group:water"}) == nil
            then
                return
            end
        end
    end
})
```

Мы ещё углубимся в API Lua в одном из будущих учебников **LXF**, но, надеемся, это было приятным введением в чудеса *Minetest*. И да подарит вам ваш блочный мир всё для жизни и процветания. **LXF**

► Наш герой и его барашек наслаждаются при свечах вкусом приготовленной крысы.



Хаос мультиплея

Как и *Minecraft*, *Minetest* состоит из клиента и сервера. При игре в однопользовательском режиме ваша машина настроена как локальный частный сервер, к которому и подключается клиентская программа. От сервера зависят все игровые настройки, включая то, какие моды активны. Желая взаимодействовать с другими игроками *Minetest*, выберите любой сервер из списка на <http://minetest.net/servers>. Они работают в различных конфигурациях: некоторые требуют пароль, некоторые работают в творческом режиме, а некоторые будут ограничивать

привилегии, такие как возможность общаться, телепорт или использование команды **/give**.

Можете запустить свой собственный сервер — либо на локальном компьютере, либо на удаленном сервере. По умолчанию он будет слушать UDP-соединения на порте 30000, так что нужно будет соответственно настроить брандмауэр. Стартуется выделенный сервер следующим образом:

```
$ minetest --server
```

Если у вас несколько миров, надо сообщить *Minetest*, какой именно использовать — либо

указав путь аргументом **--world**, либо по имени, с **--worldname**. Данные о мире хранятся в подкаталогах **~/minetest/world**. Установить всякие параметры сервера можно в файле настройки **~/minetest/minetest.conf**. Например, чтобы рекламировать свой сервер, установите параметры

```
server_announce = 1
serverlist_url = servers.minetest.net
```

Сверьтесь с файлом-образцом в **/usr/share/minetest/doc/** для полной и окончательной завершенности.



По рецептам доктора Брауна

Д-р Крис Браун

Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Учимся печатать

Одна из тех вещей, которые поражают меня в IT-профессионалах, с которыми я встречаюсь — то, что лишь совсем немногие из них по-настоящему учились печатать. Они тратят большую часть своего времени на работу с компьютером через клавиатуру, но многие лишь тыкают в нее двумя пальцами, напряженно ища нужные клавиши. Точность их набора тоже часто оставляет желать лучшего, и я иногда пугаюсь мыслей о том, как они сидят в командной строке рабочего сервера с правами root.

Я только что измерил свою скорость печати и могу заявить, что печатаю около 50 слов в минуту, если брать текст из головы. («Слово» в терминах скорости печати означает пять нажатий на клавиши.) Средняя профессиональная машинистка обычно печатает 50–100 слов в минуту. Моя скорость значительно падает (около 30–40 слов в минуту) при наборе с напечатанного текста.

Но есть огромная разница между набором английской прозы и набором причудливых команд оболочки или кода, в которых приходится сражаться со всеми видами небуквенных символов, большими буквами посреди имени переменной, необходимостью заботиться об отступах и т.д. Например, при наборе кода на Perl мне удалось достичь только 20 слов в минуту. И что немаловажно, цена потенциальных ошибок здесь гораздо выше. Например, представьте себе разницу между

```
$ rm *.old
```

и

```
$ rm *.old
```

И мне интересна ваша точка зрения. Быстро ли вы печатаете? Как вы думаете, ограничивают ли вашу продуктивность как сисадмина плохие навыки печати или плохая точность? Или время, которое уходит на придумывание набираемого текста, превосходит время на сам набор текста? Напишите мне об этом: мне будет очень интересно ваше мнение.

chris.linuxformat@gmail.com

Эзотерическое системное администрирование из причудливых заворотов кишок серверной



Cray изгнал IBM из Met Office

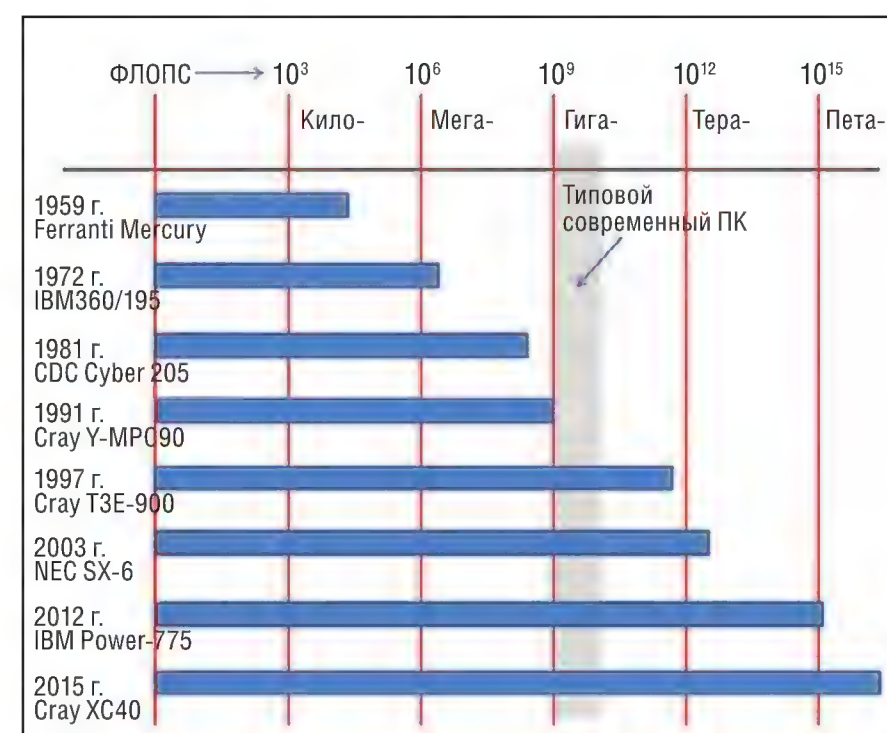
Привнося новое значение в термин «облачные вычисления», Met Office закупает такты процессора.

Met Office, национальное бюро погоды Великобритании, отваливает £97 млн на новый суперкомпьютер — Cray XC40 с разгонными полосами и титановыми дисковыми ободами. На нем, почти само собой разумеется, Linux. На сайте Cray написано, что «окружение Linux в Cray... включает ОС на базе Linux, предназначенную для запуска больших сложных приложений и эффективного масштабирования более чем на 500 000 процессорных ядер. Ядро этого окружения можно подстраивать к различной нагрузке». Детали см. на <http://bit.ly/CraySpecs>.

Met Office не новичок в суперкомпьютерах; свой первый (Ferranti Mercury) они купили еще в 1959 г. Он мог выполнять 30 000 вычислений в секунду. Для сравнения, современный компьютер быстрее раз в 100 000. С тех пор бюро приобретало не менее семи суперкомпьютеров от IBM, CDC, Cray и NEC. Новый компьютер, который приступит к работе в сентябре, будет иметь 480 000 процессорных ядер (Pentium Xeon) и заявленную производительность в 16 петафлопс. Я спросил жену, что такое петафлоп, и она сказала — наверное, это тот период позднего лета, когда цветы начинают увядать; но на самом деле «пета» означает 10^{15} , число, неправдоподобно большое настолько, что пытаюсь его представить, рискуешь нажать главную боль. У меня достаточно большой опыт работы с параллельными компьютерами, и я знаю, что они вряд ли приблизятся к этим показателям с точки зрения полезных вычислений. Но это

большое число, и этот компьютер в 13 раз мощнее IBM Power 755, который стоит в Met Office сейчас. В списке самых мощных компьютеров в мире (www.top500.org/lists) на момент написания статьи этот компьютер был четвертым.

Чем оправдываются затраты на такую мощь? Ну, Met Office утверждает, что путем «улучшения адаптации к сложным погодным условиям и сопутствующим угрозам» за счет предоставления более частых прогнозов погоды бюро заработает £2 миллиарда «социально-экономических выгод» для Великобритании, а стратегически важные объекты, такие как аэропорты, будут обеспечены прогнозами по скорости ветра, туману и снегу с пространственной точностью до 300 метров.



➤ Последние 50 лет в Met Office наблюдается экспоненциальный рост вычислительной мощности.

Настраиваем сервер OpenLDAP

Доктор показывает, как сделать информацию об учетных записях централизованной, настроив сервер OpenLDAP в Ubuntu.

Сегодняшнее утреннее чтение взято из Книги Тукса, глава пятая, стих первый. Вначале был файл паролей, и файл паролей был от Unix. Через него все входили в систему; и никто, кто прежде вышел, не мог без него войти снова. И сисадмины видели, что это хорошо. Но вдруг пришло время Великой Сети, и сказали друг другу сисадмины: «Файл паролей более не служит нам как прежде, ибо ему потребна репликация данных и он не масштабируется для великих сетей». И сказал Солнце [Sun]: «Не бойтесь, мы посылаем вам Желтые страницы, в которых данные о пользователях централизованы».

Но пришли мудрые люди из Битии, говоря: «Негоже вам брать название „Желтые страницы“, ибо оно зарегистрировано как наша торговая марка». И сказал Солнце: «Отныне то, что было известно как Желтые страницы, нарекается NIS». И сисадмины увидели, что это хорошо.

Но со временем сисадмины вновь возроптали, и пожаловались они еще раз, говоря: «Воистину, в NIS есть только плоское пространство имен и нет управления доступом».

И сказал Солнце: «Не бойтесь, мы посылаем вам NIS+, где пространство имен иерархическое и контроль доступа в изобилии». Но сисадмины возроптали в третий раз, ибо не уразумели, что сказал Солнце.

И был создан великий консорциум, породивший спецификацию X.500. И X.500 родил DAP, и DAP родил DIXIE, и DIXIE родил LDAP. И сисадмины увидели, что это хорошо.

На этом заканчивается сегодняшнее утреннее чтение.

В данной статье (отбросив библейский стиль) мы познакомимся с основами LDAP и посмотрим, как настроить сервис каталога LDAP для хранения учетных записей пользователя. В следующем месяце мы узнаем — среди прочего — как сделать так, чтобы компьютер получал информацию об учетных записях с сервера LDAP.

Букварь LDAP (поверхностно)

LDAP — сокращение от “Lightweight Directory Access Protocol [Облегченный протокол доступа к каталогам]”, но обычно, говоря о LDAP, мы также имеем в виду сервер, который общается по этому протоколу и хранит информацию в каталоге. В принципе, в LDAP можно хранить любую информацию, но на практике он чаще используется как большая адресная книга с именами пользователей, паролями, почтовыми и электронными адресами, должностями, отделами и т.д. В частности, она может содержать информацию об учетных записях пользователей — ту, что обычно хранится в `/etc/passwd` и `/etc/shadow`.

В каталоге LDAP информация хранится в виде дерева, как в файловой системе (или, если угодно, в DNS). Это дерево называется DIT (Directory Information Tree — дерево с информацией о каталоге). Каждая запись дерева идентифицируется «уникальным именем» — чем-то вроде `uid=mary,ou=People,dc=example,dc=com`. Первая часть этого имени (`uid=mary`) называется относительным уникальным именем, а остальное — уникальным именем родительского узла (`ou=People,dc=example,dc=com`). Можно провести грубую аналогию с полным путем и именем файла в файловой системе Linux, например, `/home/chris/articles/ldap`, где `ldap` — имя файла, а `/home/chris/articles` — путь родительского каталога. Но обратите внимание, что компоненты находятся в обратном порядке: в уникальных именах они записываются от младшего

Хранение данных на сервере

LDAP (как протокол) определяет способ обращения к данным, но не указывает, как они должны храниться. По умолчанию используется `hdb`, одна из разновидностей почтенной индексируемой базы данных Berkeley DB. Сами файлы по умолчанию

находятся в `/var/lib/ldap`, но их просмотр напрямую ничего не даст. Также можно употребить текстовый формат LDIF; такое применяется для дерева с атрибутом `cn=config`, но брать его для большого каталога вам вряд ли захочется.

к старшему, а в путях — от старшего к младшему. (Другое сравнение — имена DNS, например, www.sheffield.ac.uk, тоже записываются от младшего к старшему).

Уникальное имя записи верхнего уровня каталога (в нашем примере — `dc=example,dc=com`) называется контекстом именования каталога и обычно основано на DNS-имени вашей организации (`example.com`), так как оно гарантированно является уникальным. Устанавливать контекст именования в `dc=com` нельзя, так как в этом случае в нашем каталоге окажутся данные для всего домена `.com`!

Каждая запись каталога по сути представляет собой набор атрибутов и значений. Вскоре мы создадим запись для пользователя **mary**, которая (среди много другого) включает следующие атрибуты:

`uid: mary`

`sn: Brown`

`givenName: Mary`

Атрибуты немного похожи на переменные в языках программирования, где мы можем сказать, что `uid` — переменная со значением **mary**. Но не уводите эту аналогию слишком далеко, потому что, в отличие от переменных, у атрибутов может быть несколько значений. Например, у **mary** мы можем увидеть следующее:

`telephoneNumber: 01263 987654`

`telephoneNumber: 07639 123456`

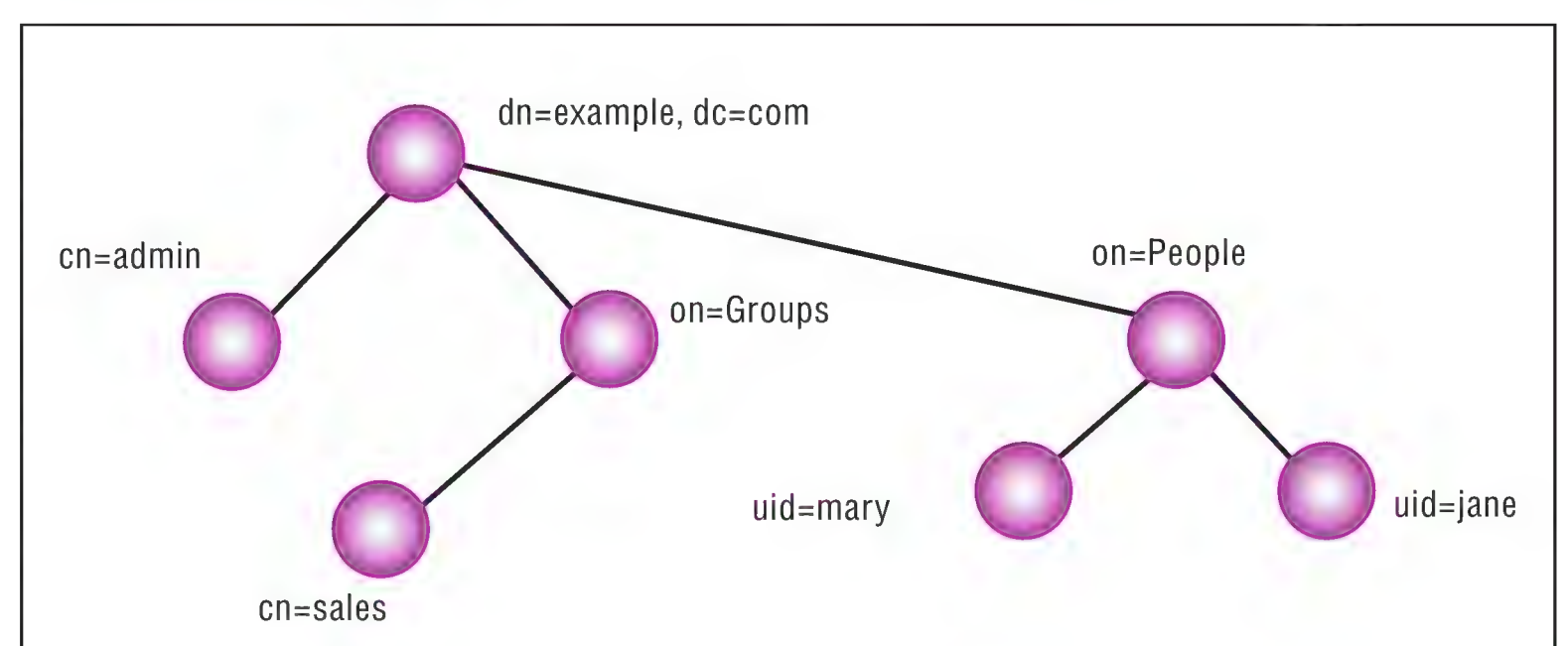
так как у настоящей Мэри два телефона.

У каждой записи в каталоге должен быть атрибут, называемый **objectClass**. Значение этого атрибута определяет шаблон, который, в свою очередь, определяет обязательные и необязательные атрибуты этой записи. Таким образом, дополнив запись для пользователя **mary**, мы можем получить что-нибудь такое:

`dn: uid=mary,ou=People,dc=example,dc=com`

`objectClass: inetOrgPerson`

»



» Дерево с информацией о каталоге LDAP, полученного в этой статье.


```
objectClass: posixAccount
objectClass: shadowAccount
uid: mary
sn: Brown
givenName: Mary
```

Здесь мы видим, что **objectClass** Мэри содержит три значения: **inetOrgPerson**, **posixAccount** и **shadowAccount**. Сочетание этих значений определяет, какие атрибуты обязательно должны быть у записи **mary** и какие необязательны. Программисты, знакомые с ООП, могут сравнить эти определения **objectClass** с классами, и действительно — классы объекта можно наследовать от других классов; так, вы обнаружите, что **inetOrgPerson** наследуется от **organizationalPerson**, который наследуется от **person**. Эти определения классов образуют так называемые схемы каталога.

Но я рискую запугать вас всей этой теорией. На практике, если вы хотите сделать информацию об учетных записях централизованной с помощью LDAP, слишком глубоко залезать в дебри нежелательно. Так что продолжим...

Настройка LDAP

Наша задача в этом месяце — просто настроить сервер LDAP для хранения информации о учетных записях пользователей. Для этого я пользуюсь Ubuntu 14.04. Основные моменты для всех дистрибутивов Linux одинаковы, но подробности будут отличаться. Я — данная тема в моде — уже про это писал [см. «Рецепты доктора Брауна» **LXF187**, стр. 61], показав, как с помощью *Winbind* получить информацию о пользователях из Active Directory, в сердце которой, разумеется, LDAP. Но на сей раз попробуем реализовать наш собственный сервис LDAP.

Мы установим сервер OpenLDAP, под названием *slapd*, который автоматически настроит сам себя с минимальным участием с нашей стороны. Однако этот сервер берет контекст именования (имя записи верхнего уровня для каталога) из имени хоста компьютера, поэтому нужно начать с включения имени хоста в **/etc/hosts** — например, так:

```
127.0.0.1 localhost.example.com localhost
127.0.1.1 chris-hp250.example.com chris-hp250
```

```
Далее установим пакеты:
$ sudo apt-get install ldap-utils

Эта команда установит полдюжины клиентских утилит с их man-страницами. Затем установим сервер:
$ sudo apt-get install slapd
```

Вас попросят задать пароль для учетной записи администратора LDAP: **cn=admin,dc=example,dc=com**. Пакет *slapd* содержит собственно сервер LDAP и ряд вспомогательных утилит с их man-страницами, набор схем и несколько вспомогательных библиотек.

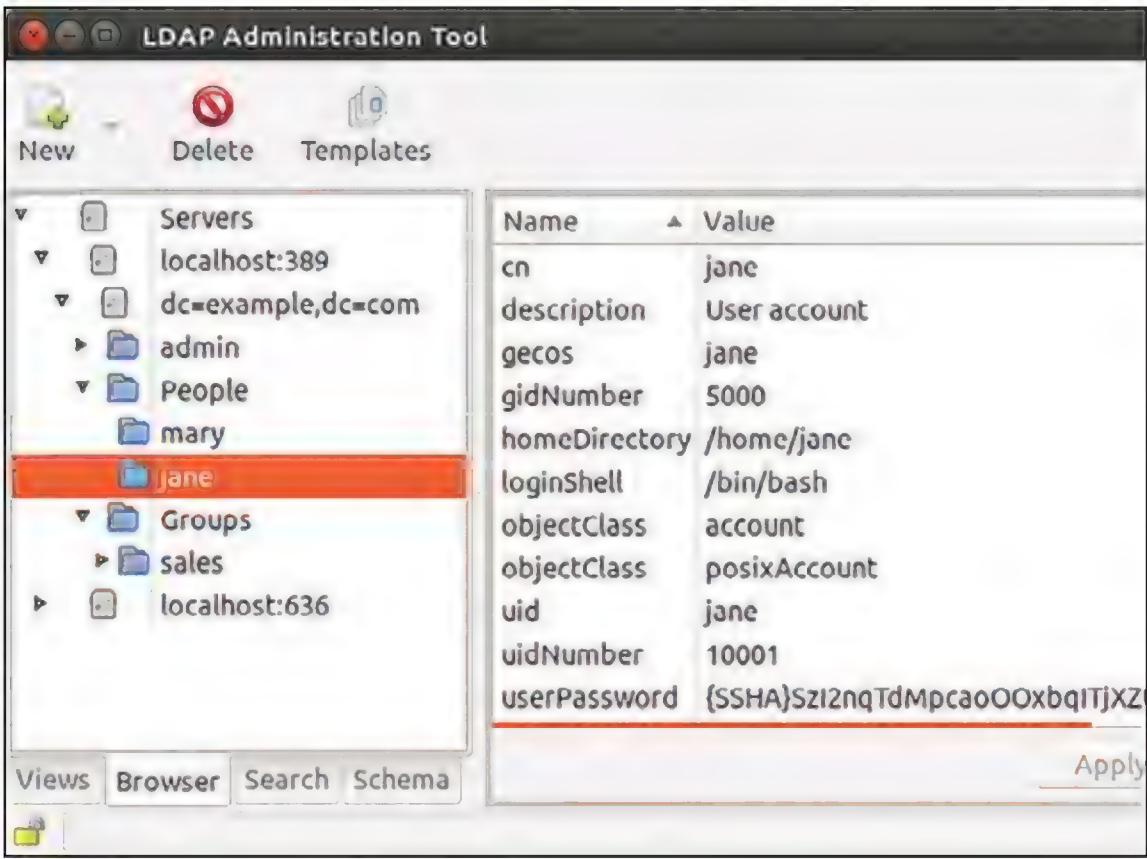
Обычно у сервера должен быть файл настройки, например, **/etc/slapd.conf**. Хотя в ранних версиях OpenLDAP он был, теперь информация о конфигурации перенесена в собственное дерево (DIT). Файлы LDIF, из которых загружается это дерево, хранятся в каталоге **/etc/ldap/slapd.d/cn=config**. Учтите, что эти файлы нельзя редактировать вручную. Подробности см. на <http://bit.ly/OpenLDAPAdminGuide>.

Создаем пользователя

Как обычно в дистрибутивах, производных от Debian, при установке сервиса сервер автоматически настраивается и переводится в минимальное рабочее состояние. Поэтому мы, недолго думая, можем добавить некоторое содержимое. Вот как это сделать сложным способом. Сначала создайте файл **populate.ldif** со следующим содержимым:

```
dn: ou=People,dc=example,dc=com
objectClass: organizationalUnit
ou: People
dn: ou=Groups,dc=example,dc=com
objectClass: organizationalUnit
ou: Groups
dn: cn=sales,ou=Groups,dc=example,dc=com
objectClass: posixGroup
cn: sales
gidNumber: 5000
dn: uid=mary,ou=People,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: mary
sn: Brown
givenName: Mary
cn: Mary Brown
displayName: Mary Brown
uidNumber: 10000
gidNumber: 5000
userPassword: marybrown
gecos: Mary Brown
loginShell: /bin/bash
homeDirectory: /home/mary
```

➤ **LAT** (утилита администрирования LDAP) — одна из нескольких графических утилит для просмотра, поиска и управления содержимым каталога LDAP.



Каталоги и базы данных

И каталоги типа LDAP, и базы данных типа *MySQL* предоставляют высокоструктурированный подход к хранению и загрузке данных. Но каталоги и базы данных очень отличаются друг от друга. Во-первых, данные в LDAP существуют в древовидной структуре — она иерархична. Ветви дерева не могут иметь никаких «связей» друг с другом. А в базах данных информация хранится в таблицах, и можно задать отношения между этими

таблицами через первичные/внешние ключи. Схема LDAP (определяющая типы атрибутов, которые могут появляться внутри узла) аналогична схеме таблиц баз данных (с именами и типами столбцов), но «объединить [join]» (в реляционном смысле), но две части каталога LDAP никак нельзя. Еще одно отличие — каталоги «преимущественно предназначены для чтения». Обычно затраты на обновление элемента каталога гораздо выше, чем на чтение

этого элемента. Так, в самом крайнем случае для добавления одного пользователя NIS нужно перекомпилировать всю карту паролей. Другой крайний случай — печатными телефонными книгами ежедневно пользуются тысячи подписчиков. Для обновления такой книги (обычно раз в год) нужно напечатать и разослать массу бумаги. С базами данных обновления выполняются чаще, и частота чтения и записи сбалансирована лучше.

Это пример файла LDIF. LDIF — текстовый формат, дающий «внешнее» представление содержимого каталога LDAP. В этом файле описаны две важных записи верхнего уровня: **People** и **Groups**. Под ними мы добавляем группу **sales** и пользователя **mary**. Наше дерево каталога показано на рисунке на стр. 59. Создав этот файл, добавьте его в каталог следующим образом:

```
ldapadd -x -D cn=admin,dc=example,dc=com -W -f populate.ldif
```

Вас попросят указать пароль, заданный для учетной записи администратора LDAP при установке сервера.

Учтите, что при такой установке пароля для **mary** он хранится в каталоге в текстовом виде; это плохая идея. Так или иначе, подготавливать LDIF вручную — явно не самый удобный способ управления учетными записями пользователей, поэтому давайте рассмотрим средства более высокого уровня.

Во-первых, в пакете *ldapscripts* есть набор скриптов оболочек, которые оборачиваются вокруг стандартных утилит командной строки для более удобного управления учетными записями и группами. Установить его можно командой

```
$ sudo apt-get install ldapscripts
```

В файле настройки */etc/ldapscripts/ldapscripts.conf* следует задать контекст именования каталога (в нашем случае это **dc=example,dc=com**) и, возможно, еще несколько параметров. Основные параметры, которые я изменил, таковы:

```
SUFFIX="dc=example,dc=com"
GSUFFIX="»ou=Groups»"
USUFFIX="»ou=People»"
MSUFFIX="»ou=Machines»"
BINDDN="»cn=admin,dc=example,dc=com"
```

В последней строке указана учетная запись, которую я буду использовать. Также нужно указать пароль администратора LDAP в файле */etc/ldapscripts/ldapscripts.passwd* (имя этого файла задается параметром **BINDPWDFILE** в файле *ldapscripts.conf*) таким образом:

```
sudo echo -n mysecret > /etc/ldapscripts/ldapscripts.passwd
sudo chmod 400 /etc/ldapscripts/ldapscripts.passwd
```

Команда **echo -n** важна; символ новой строки в конце файла приведет к тому, что файл не будет работать. Теперь можно добавить пользователя в каталог гораздо проще, вот так:

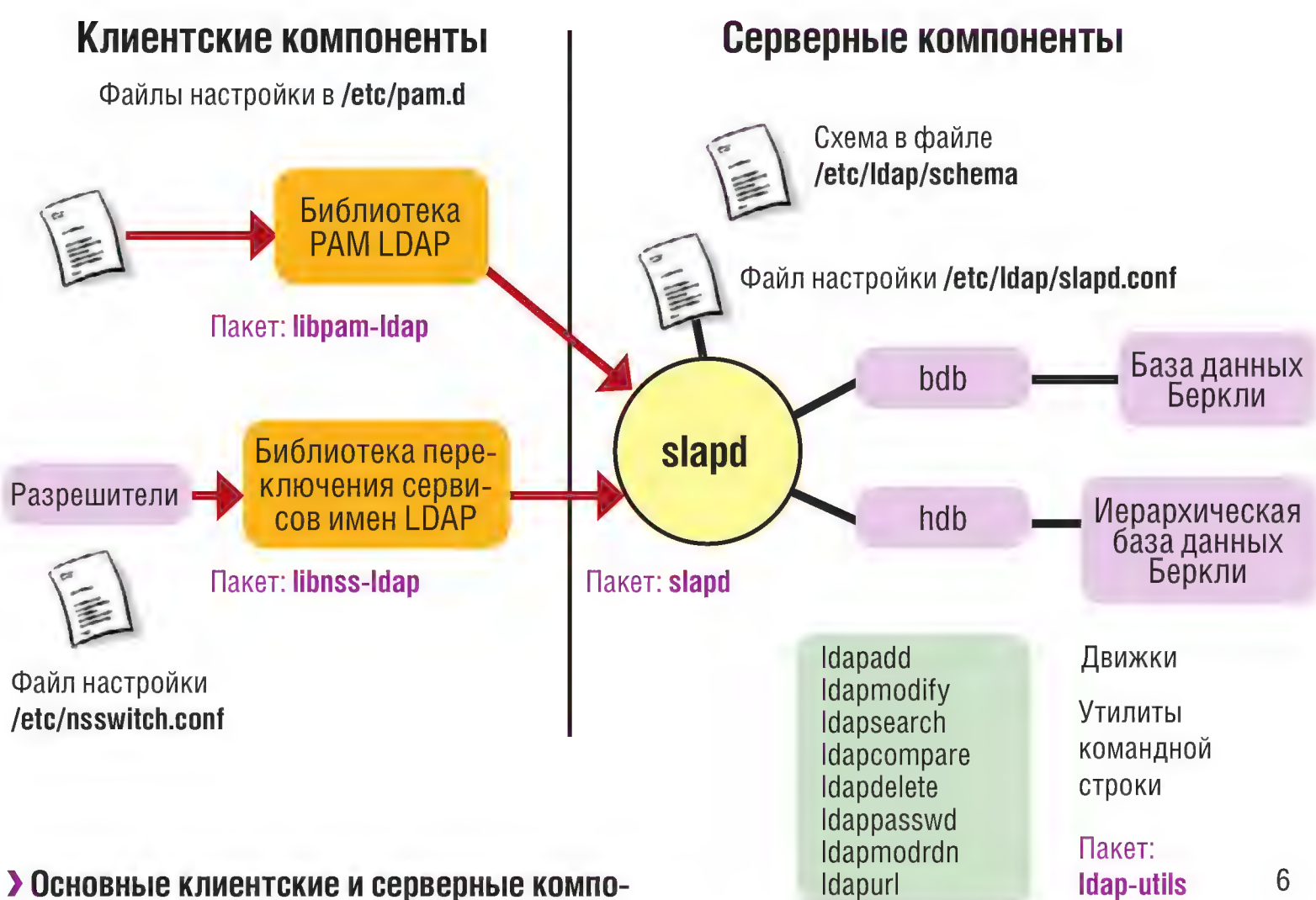
```
ldapadduser jane sales
Successfully added user jane to LDAP
Successfully set password for user jane
```

Если не получилось, поищите подсказки в файле */var/log/ldapscripts.log*. Пакет *ldapscripts* включает еще несколько полезных скриптов: *ldapaddgroup*, *ldapsetpasswd*, *ldapmodifyuser* и т.д. Если вам требуются простые утилиты для управления учетными записями пользователей в LDAP, этих скриптов должно быть достаточно.

Как выполнять поиск

Каталоги предназначены для поиска информации. В командной строке можно воспользоваться *ldapsearch*. Здесь мы ищем числовой идентификатор пользователя **jane**:

```
$ ldapsearch -x -LLL -b dc=example,dc=com 'uid=jane' uidNumber
dn: uid=jane,ou=People,dc=example,dc=com
uidNumber: 10001
```



➤ Основные клиентские и серверные компоненты OpenLDAP. То, что выделено оранжевым, мы обсудим через месяц.

Здесь мы начинаем поиск с **dc=example,dc=com** (вершина дерева каталога) и ищем записи, у которых атрибут **uid** имеет значение **jane**. Для каждой найденной записи (на самом деле, будет всего одна) мы выводим атрибут **uidNumber**. При желании можно вывести несколько атрибутов:

```
$ ldapsearch -x -LLL -b dc=example,dc=com 'uid=jane' uidNumber loginShell
```

или все атрибуты для найденной записи:

```
$ ldapsearch -x -LLL -b dc=example,dc=com 'uid=jane'
```

Наведи-и-щелкни

Если вы предпочитаете графические утилиты, то для вас имеется несколько утилит, помогающих просматривать содержимое и управлять каталогом LDAP. Одна из таких утилит — *LAT* (LDAP Administration Tool — утилита администрирования LDAP); она есть в репозиториях Ubuntu, поэтому установить ее легко:

```
sudo apt-get install lat
```

Обратите внимание, что это приложение .NET, и при установке оно потянет за собой среду выполнения Mono, если у вас ее еще нет. *LAT* — отличная утилита для управления пользователями и группами, просмотра каталога и поиска в нем, а также просмотра схем, и я бы настоятельно рекомендовал провести с ней некоторое время. Другие графические утилиты, достойные внимания, включают *phpLDAPadmin* (с web-интерфейсом) и *Apache Directory Studio*.

На этом заканчивается первая часть нашего руководства. Мы настроили каталог LDAP, пригодный для централизованного хранилища учетных записей. Через месяц я расскажу, какие изменения нужно сделать на клиентской стороне, чтобы реально работать с каталогом. Я также планирую коснуться репликации каталога, чтобы избежать единственной точки отказа и/или распределить нагрузку. Мы также рассмотрим, как настроить шифрованное соединение с сервером, применив SASL. **LXF**

Облегченный протокол?

Обозрев сложности протокола LDAP, вы можете спросить себя, почему он называется облегченным. Действительно, облегченный по сравнению с чем? Ну, корни LDAP лежат в сервисе каталогов X.500, гораздо

более сложном фреймворке; он появился в 1988 году и предназначался для запуска поверх стека протоколов OSI, который представлял собой семиуровневую модель, активно продвигавшуюся в то время, пока

не победило благоразумие и ведущую роль не стал играть TCP/IP. LDAP изначально задумывался как легкий протокол, который выступал бы шлюзом для запросов к серверам X.500.

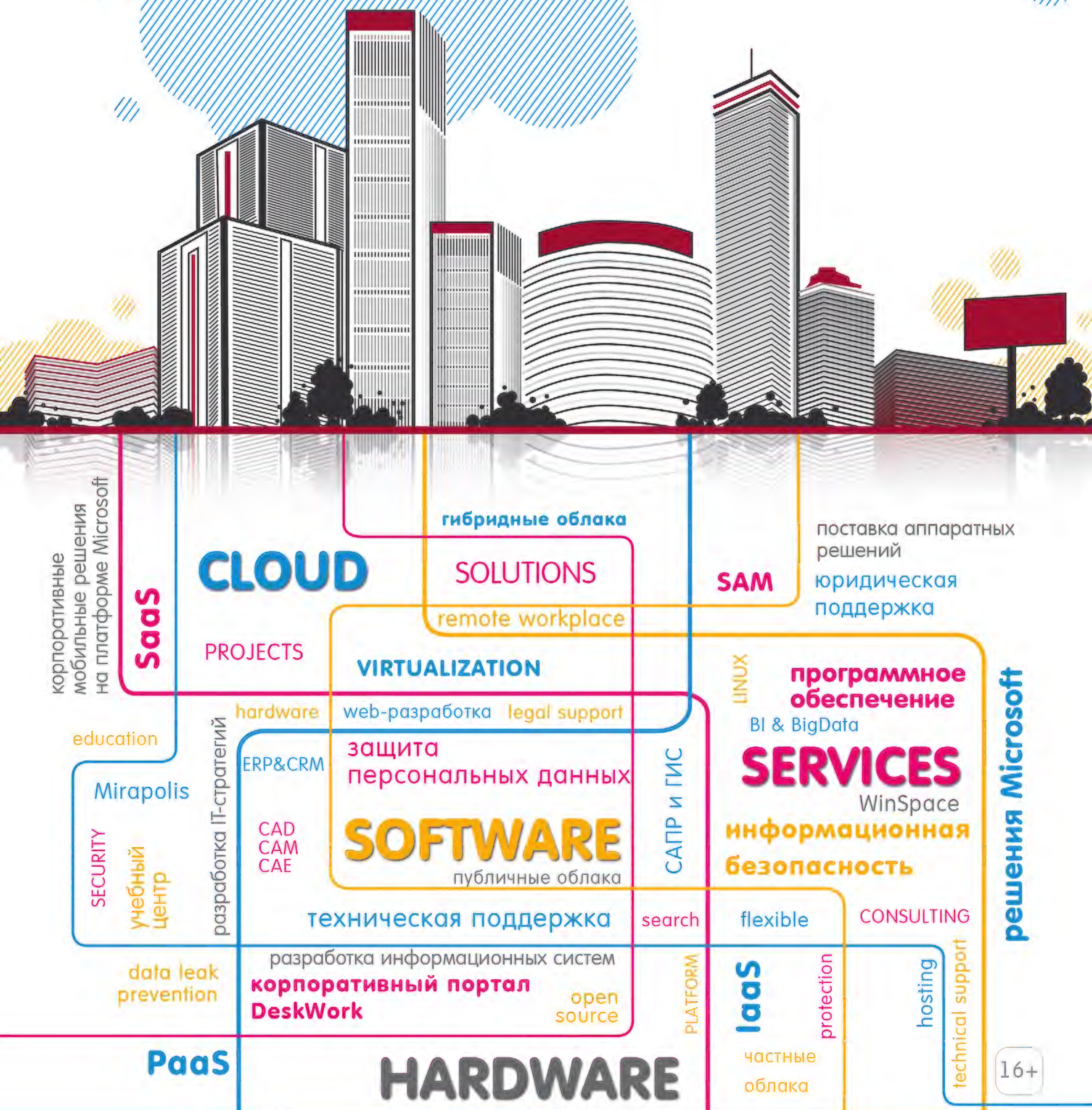
softline®

Cloud Software Hardware Services

20+

Years in IT

IT-архитектура вашего бизнеса



Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Телевизоры умнеют

Скоро единственной причиной включить телевизор будет желание посмотреть на Linux в нём.
Объективная реальность

В прошлом году компания LG начала выпускать телевизоры на WebOS. В этом году версия WebOS доросла до 2.0 и вошла в новую линейку ТВ LG, теперь «ультра-высокого разрешения». А Samsung в самом начале года заявила, что все её «Smart TV» будут на Tizen.

Обе эти ОС построены вокруг ядра Linux и имеют мобильные корни, отчасти выросшие из ныне покойных легенд в мобильных делах Palm и Nokia. Обе вначале оптимизировались для управления маленьким кирпичиком, который по идее помещался в кармане. А теперь угодили на ТВ. Это как оптимизированный под малолитражку движок запихать в самосвал. Самое забавное — в случае WebOS и Tizen это сработало.

Linux уже прочно захватил ТВ — конечно, только те, которые из-за наличия «мозгов» прозываются «умными». Но в этой области велика фрагментация прошивок, а разработчики не считают зазорным тивоизировать свои устройства. А значит, об обновлении ПО, исправлении ошибок и добавлении функций смело забудьте. Применение готовых открытых дистрибутивов в домашней электронике крупными компаниями должно снизить фрагментацию и упростить модификацию прошивок и разработчикам, и энтузиастам.
PS Google добавил в свои Docs поддержку *Open Office*. Кривовато, поздно, но это лучше, чем ничего.
E.M.Baldin@inp.nsk.su

В этом месяце вы научитесь...



Преображать Grub 64

Нейлу Ботвику симпатичен сам *Grub*, но не симпатичен его скучноватый вид, и он берется за переделку, заодно подправляя пункты меню.



Изучать микроядро 66

Спор Торвальдса с Таненбаумом к окончательному решению не привел. **Ричард Смедли** в поисках истины обращается к ОС Minix.



Стирать данные 70

На зависть параноикам, **Шон Конвэй** уничтожает персональные данные на своем диске, иной раз даже впадая в крайности. Зато никто не сумеет проникнуть в его секреты!



Упрощать брандмауэр 74

Даниил Батурин применяет *ipset*, и — вот чудо-то! — считывать и модифицировать замысловатые конфигурации становится на удивление легко.



Обрабатывать текст 78

Системные журналы хладнокровно регистрируют все подряд и без разбора, поэтому **Эндрю Моллетт** принялся просеивать их с помощью *sed* и *Awk*.



Зажигать на празднике 82

Михаила Остапкевича и **Евгения Балдина** настолько заворовили новогодние елки, что они решили продлить очарование сияющих огоньков.

АКАДЕМИЯ КОДИНГА



И новичкам, и гуру!
Всегда полезно будет познать
нечто доселе неведомое

Cython и Уолш с Адамаром 88

Джонни Бидвелл временно отвлекся от ядер операционных систем, чтобы обработать снимки ядра кометы 67P, добытые орбитальным модулем миссии «Розетта».

R — язык статистики 92

Мониторинг системы — конек **Михалиса Цукалоса**. Но ведь накопленные данные надо еще и обработать, и визуализировать... Хорошее подспорье тут представляет язык R.

Grub: Личное меню загрузки

Нейл Ботвик расскажет, как переделать довольно унылое меню загрузки по умолчанию, чтобы оно выглядело и действовало так, как вы пожелаете.



Наш эксперт

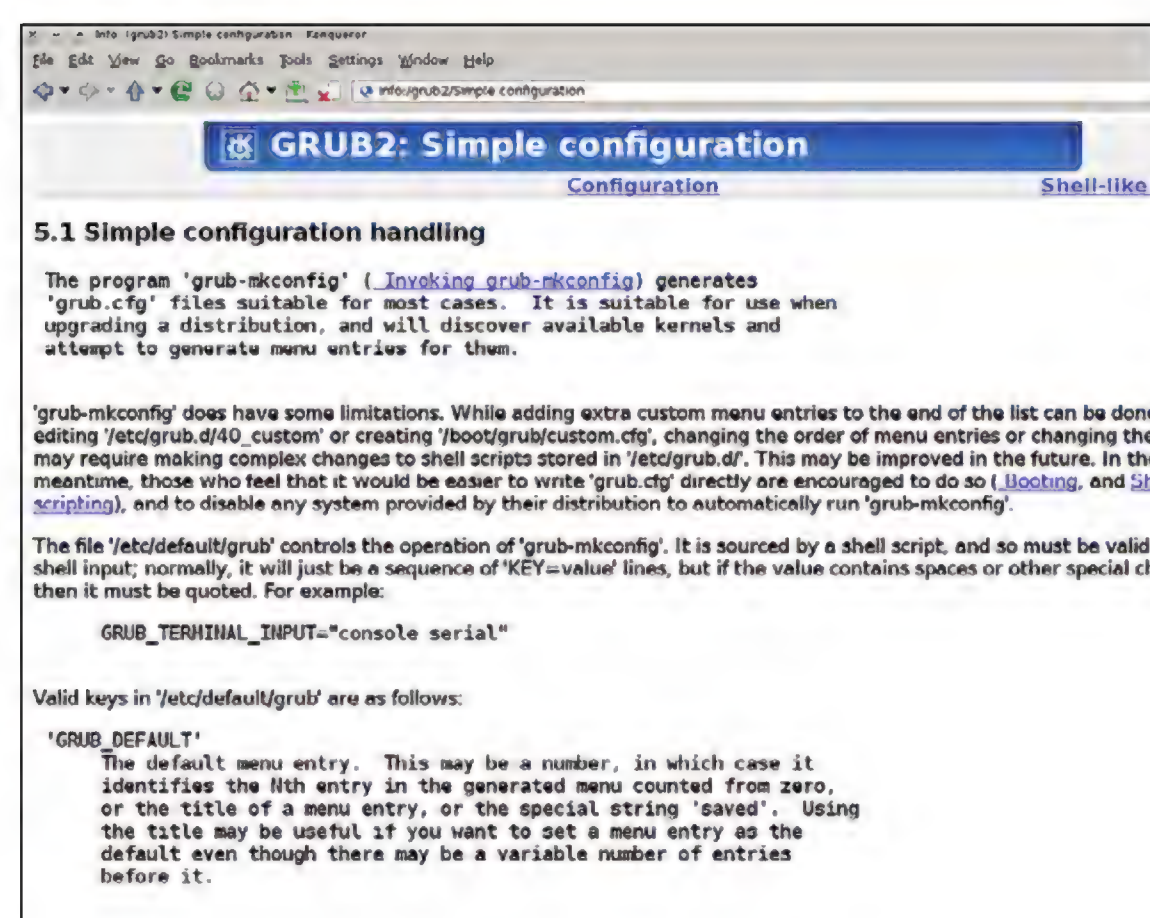
У Нейла Ботвика по компьютеру в каждой комнате, но по соображениям безопасности он нипочем не расскажет, где находится центральный сервер.

Загрузки наподобие *Grub* — из ряда тех программ, которые особо не замечаешь, пока не возникают проблемы. Вы включаете компьютер, появляется (может быть) меню загрузки, а потом загружается система. Между тем, в этом меню есть множество нюансов, определяющих, как именно будет загружаться ваш компьютер, а также доступных опций.

На данный момент в ходу две основных версии *Grub*. Текущая версия называется *Grub 2*, а первоначальную часто именуют *Grub Legacy*, поскольку версии 1.0 она так и не достигла: последней, которую вы найдете, будет 0.97. В *Grub Legacy* был файл настройки — `/boot/grub/menu.lst`, который нужно было редактировать для добавления или удаления опций. Для тех, кто легко читает текстовые файлы настройки, всё было прекрасно, но для автоматической генерации меню загрузки оно было не очень удобно, поэтому в *Grub 2* используется другая система. Там тоже есть файл настройки в `/boot/grub`, который теперь называется `grub.cfg`, но этот файл можно создать автоматически — и, как правило, на основе того, что установлено на вашем компьютере. Мы рассмотрим, как повлиять на этот процесс, чтобы настроить меню загрузки под ваши потребности, путём добавления опций, ускорения или создания более привлекательного меню для менее технически-подкованных пользователей.

Как устроено меню

Основной файл настройки расположен в `/boot/grub`, но его, как правило, автоматически генерирует `grub-mkconfig` (у пользователей Ubuntu есть `update-grub` — скрипт, который вызывает `grub-mkconfig`). При запуске `grub-mkconfig` обращается к двум источникам, чтобы определить, как ему действовать: `/etc/default/grub` и `/etc/grub.d`. В первом файле содержатся некоторые переменные окружения, которые использует *Grub*, а в `/etc/grub.d` — ряд



» Все нужные вам настройки, и даже более того, подробно описаны на страницах `info`.

скриптов оболочки, выполняемых для создания файла настройки. Самые важные скрипты — `00_header` и `10_linux`. Первый определяет конфигурацию *Grub*, создавая первую часть `grub.cfg` с глобальными настройками. Затем `10_linux` сканирует жесткий диск на наличие систем Linux и создает записи меню для каждого ядра в каждом установленном дистрибутиве. Если у вас параллельно установлена Windows, `30_os-prober` затем добавляет пункты меню для операционных систем, отличных от Linux. Эти скрипты предусмотрены в *Grub*, и менять их незачем, но вы можете повлиять на образ их действий, устанавливая или изменяя значения переменных в `/etc/default/grub`.

Смена настроек по умолчанию

Первым делом вы, возможно, захотите изменить параметр `GRUB_TIMEOUT`, который устанавливает, сколько секунд отображается меню перед загрузкой варианта по умолчанию. Если вы почти всегда выбираете именно его, попробуйте уменьшить значение до 2 или 3, чтобы ускорить этот процесс. Если установить 0, *Grub* будет загружать вариант по умолчанию немедленно, но тогда параметры восстановления или другие системы станут вам недоступны. Установка в -1 полностью отключит таймаут. `GRUB_HIDDEN_TIMEOUT` выполняет аналогичную функцию, если меню скрыто, как в некоторых дистрибутивах. Он определяет, как долго будет длиться пауза перед загрузкой, в ожидании нажатия Esc для открытия меню. Закомментируйте его, если хотите, чтобы меню отображалось всегда. Если вы не используете Windows, можно сэкономить немного времени при создании меню, установив

```
GRUB_DISABLE_OS_PROBER=true
```

Перемените тему

Мы рассмотрели только то, как изменить содержание меню загрузки, однако *Grub* позволяет также изменить и свой внешний вид. Вы можете установить графический фон, указав в `GRUB_BACKGROUND` полный путь к файлу PNG или JPEG, или выбрать тему оформления для всего, что отображается на экране.

Темы содержатся в каталогах в `/boot/grub/themes`, и вы можете выбрать одну из них, указав `GRUB_THEME` и имя папки (не полный путь) в `/etc/default/grub`.

По умолчанию установлена тема *Starfield*, хотя в некоторых дистрибутивах она упакована отдельно, так что вам, возможно, потребуется её установить. Темы устанавливаются в `/usr/share/grub/themes`. Скопируйте те, что хотите использовать, в `/boot/grub/themes`, обозначьте нужную как `GRUB_THEME` и перезагрузите компьютер. Если вы хотите поэкспериментировать, начните с существующей и измените её. Все компоненты описаны на `info`-страницах *Grub*.

Каждый пункт меню проходит через ядро, корневую файловую систему и соответствующий `initrd`, если таковой имеется. Вы можете добавить дополнительные опции в командной строке ядра, добавляя их к `GRUB_CMDLINE_LINUX` или `GRUB_CMDLINE_LINUX_DEFAULT`. Первый охватывает все пункты меню, а второй не применяется к опциям восстановления. Некоторые не любят использовать `UUID` в `/etc/fstab` и загрузочных меню, поскольку для человека они практически нечитаемы. Вы можете отключить их с помощью

```
GRUB_DISABLE_LINUX_UUID=true
```

Существуют и другие опции, не приведённые в комментариях к файлу. За полным списком обратитесь в раздел конфигурации на info-страницах *Grub*.

Пользовательские пункты меню

Изменение настроек в `/etc/default/grub` варьирует внешний вид или поведение стандартных пунктов в меню; ну, а если вы захотите добавить свой собственный? Простейший способ сделать это — редактировать `/etc/grub.d/40_custom`. Этот скрипт, как и все остальные, служит для построения меню, но сам по себе ничего не добавляет. Вы можете дополнить его теми пунктами, которые вы хотите видеть в меню; но не удаляйте существующий контент. Например, вы можете сохранить копию ISO-файла Rescatux в `/boot` и добавить для него пункт меню, дописав в `40_custom`:

```
menuentry "Rescatux 0.32 - 64 bit" {
    isofile=rescatux-0.32b2.iso
    loopback loop $isofile
    linux (loop)/live/vmlinuz1 findiso=$isofile boot=live config quiet splash
    initrd (loop)/live/initrd1.img
}
```

Поскольку файл ISO здесь упоминается дважды, мы сделали его переменной: так будет легче обновлять. Теперь, когда вы запустите `grub-mkconfig`, у вас появится строка для Rescatux.

Скрипты в `grub.d` выполняются последовательно, и если вы хотите, чтобы ваши пользовательские пункты стояли перед опциями Windows из `30_os-prober`, просто переименуйте `40_custom` в `15_mystuff`. Количество добавляемых скриптов также не ограничено: только позаботьтесь добавить верхние строчки из `40_custom` и сделайте свой скрипт исполняемым. Вы можете запустить каждый скрипт из `/etc/grub.d` в терминале, чтобы увидеть, что именно он добавляет в меню.

Написание собственных пунктов меню вручную повышает вероятность синтаксических ошибок, поэтому всегда проверяйте правильность файла, прежде чем его запускать. Если же при загрузке всё-таки возникли проблемы, нажмите E в меню *Grub* для внесения изменений. На радость неуклюжим наборщикам, в *Grub* есть инструмент для проверки вашего меню:

```
grub2-script-check /boot/grub/grub.cfg
```

Удалить один-два пункта меню в `40_custom` легко, но не забывайте менять его всякий раз, когда вы что-либо обновляете. Поскольку скрипты в `/etc/grub.d` — это просто набор команд для `grub.cfg`, можно попробовать его усложнить. Например:

```
for ISO in /boot/rescatux*.iso; do
    echo menuentry "\Rescatux 64 bit from $(basename $ISO)" {"
    echo -e "\tisofile=$ISO"
    echo -e "\tloopback loop \${isofile}"
    echo -e "\tlinux (loop)/live/vmlinuz1 findiso=\${isofile} boot=live config quiet splash"
    echo -e "\tinitrd (loop)/live/initrd1.img"
    echo «}»
```

Grub или Grub 2?

Между дистрибутивами всё ещё есть некоторые разночтения относительно имен команд и расположения файлов. К именам команд *Grub 2* раньше было принято добавлять 2, чтобы обе версии *Grub* могли быть установлены одновременно.

Так, `grub-mkconfig` стала `grub2-mkconfig`. В некоторых дистрибутивах это сохраняется, поэтому уточните в `/usr/bin`, какие имена используются у вас. Аналогично, `grub.cfg` может находиться в `/boot/grub` или `/boot/grub2`.

done

Сохраните это в `/etc/grub.d`, сделайте его исполняемым и запустите. Меню будет выглядеть так же, но при добавлении или замене ISO новой версией оно будет адаптироваться соответственно. Конечно, вы можете полностью заменить существующие скрипты собственными, если хотите совершенно переделать установку. Но не поддавайтесь искушению изменять существующие скрипты, поскольку все ваши труды будут перезаписаны при первом же обновлении пакетов. Вместо этого, просто сделайте те скрипты, которые не хотите запускать, неисполняемыми, а своему дайте уникальное имя.

Работа с новым меню

Изменения в `/etc/default/grub` и `/etc/grub.d` не применяются к меню *Grub*, пока вы не запустите `grub-mkconfig`, который из всего, что мы рассмотрели, создаёт файл конфигурации. Лучше всё-таки сначала проверить синтаксис меню:

```
grub-mkconfig | grub-script-check
```

Если никаких ошибок не обнаружится, то конфигурацию можно без опаски добавлять в существующий файл меню:

```
grub2-mkconfig -o /boot/grub/grub.cfg
```

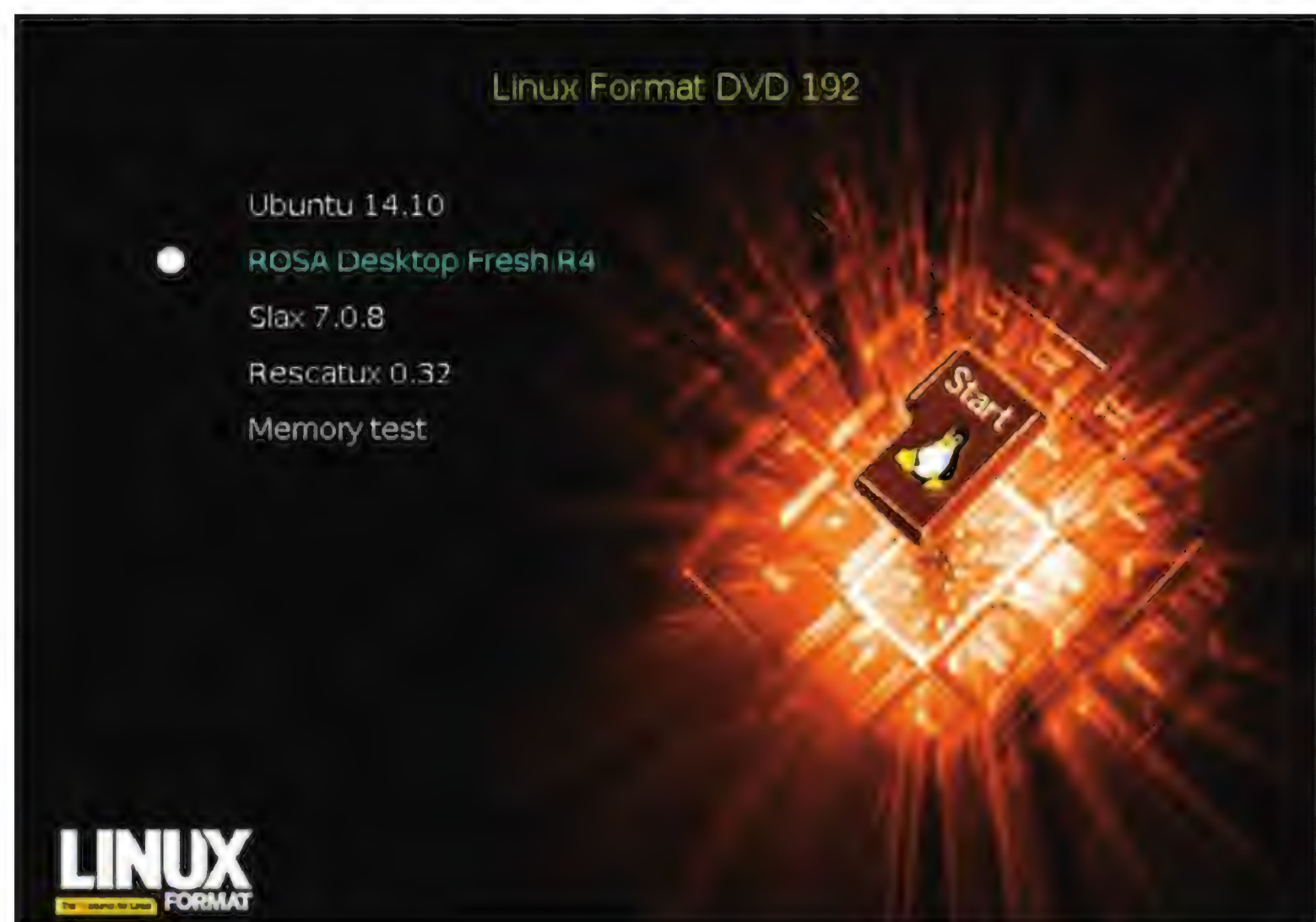
Это обычное местоположение файла меню в системах с BIOS. С UEFI он может очутиться в каком-то другом месте; в этом случае найти его поможет `locate`.

Теперь при загрузке компьютера должно появиться ваше пользовательское меню. В *Grub 2*, безусловно, больше разнообразия и гибкости. Благодаря ему вы можете сделать свой компьютер по-настоящему уникальным, прямо с момента включения. **LXF**

Скорая
помощь



Изменение параметров загрузки требует привилегий суперпользователя: вы вносите их в системные директории `/etc` и `/boot`. Команды нужно выполнять от имени `root` или с префиксом `sudo`.



» Вы можете полностью изменить облик меню загрузки, используя тему. Эту вы, наверное, уже видели.

» Подпишитесь на печатную и электронную версии на www.linuxformat.ru/subscribe!

Minix 3: Тест ОС с микроядром

Ричард Смедли обнаруживает, что с поддержкой ARM Minix 3 ушла далеко от образовательной роли; но надежность микроядра все равно у нее в сердце.



Наш эксперт

Ричард Смедли опробовал все POSIX-совместимые ОС из 1990-х, от V2_OS и BeOS до AtheOS и даже GNU Hurd.



Мinix 3 — клон Unix на основе микроядра, отличающийся высокой надежностью, отказоустойчивостью и низкими требованиями к памяти. В микроядре драйверы и серверы изолированы, работают в пользовательском режиме, а их влияние на остальную часть системы ограничено (подробнее об этом ниже). За это приходится расплачиваться небольшим снижением производительности и некоторыми сложностями с архитектурой, но для большинства пользователей производительность менее важна, чем защищенность и стабильность.

История у микроядер долгая. В 1980-х исследователи были убеждены, что это единственное направление разработки ОС, и у автора Minix Эндрю Таненбаума [Andrew Tanenbaum] и в мыслях не было проектировать Minix иначе. По этой же причине Ричард Столлмен [Richard Stallman] обратился к микроядру как к основе Hurd, ядра GNU OS. Неизбежный приход GNU — и BSD Unix — подтолкнул Таненбаума к применению Minix как чисто академической ОС, и это удачно совпало с потребностью иметь нечто достаточно компактное для обучения. Таненбаум описал это в своей знаменитой книге «Операционные системы: проектирование и реализация [Operating Systems: Design and Implementation]».

Из-за юридических проблем и технических сложностей разработка Hurd была приостановлена, но сосредоточенность на Minix позволила Таненбауму более или менее в одиночку создать Minix 1, а тридцать лет написания кода с тщательными и всесторонними исследованиями его аспирантов в Свободном университете Амстердама заложили серьезную основу для Minix 3.

Несмотря на исторические перебранки с Linux (см. «История древнего мира», стр. 67) и акценте ранних релизов Minix

на обучении, система стала привлекательной для многих пользователей, особенно потому, что постепенно впитала сообщество пользователей NetBSD. Это было достигнуто преимущественно благодаря миллионному финансированию разработки «безопасной и надежной» ОС от EC и щедрому вкладу студентов-участников программы Google — Summer of Code [Лето за кодом].

Последнюю версию Minix команда разработчиков представила на FOSDEM в этом году, и в ней охват 'MINI UNIX' расширился, но при этом система не раздулась. Например, поддержка ARM была перенесена в основную ветку, и, по словам разработчиков, версия 3 получила «более полную совместимость с NetBSD в утилитах, вызовах, типах (много 64-битных), инструментарии, базе кода и пакетах», а «весь код, привязанный к Minix, был перенесен в подкаталог верхнего уровня `minix/`».

Minix 3.3.0 еще не готова к массовому применению, но может стать стабильной основой для нескольких проектов, поэтому мы призываем вас попробовать ее, на BeagleBoard или на виртуальной машине, и следить за разработкой.

К сожалению, для тех, кто хочет сделать это не на ARM, а на старом добром настольном ПК, есть одно большое препятствие: система X Window, которая работала в предыдущем релизе (на основе монолитного сервера XFree86), картинно вышла из строя. На момент написания статьи текущий релиз X.org портировался, но еще не был готов, и пока в вашем распоряжении только командная строка. Впрочем, если вы очень хотите попробовать рабочий стол, можете загрузить предыдущие релизы. В Minix запустятся не все пакеты из NetBSD, но совместимость улучшается, а сборка из исходников может помочь получить те любимые пакеты, которых нет в репозиториях.

Внутри (виртуальной) машины

Minix 3 установится на ваш компьютер с архитектурой x86, но аппаратная совместимость системы ограничена. Любой процессор i586 или более поздней версии должны прекрасно подойти, но с Pentium 4 отмечались проблемы. Для работы системы достаточно всего 32 МБ оперативной памяти. Еще одна проблема — сетевые карты. Список совместимых сетевых карт невелик, но включает сетевые карты, эмулируемые в *Bochs*, *Qemu*, *VirtualBox*, *Virtual PC* и *VMware*, а также в LAN8701A на одноплатном компьютере BeagleBone. Остальные сетевые карты в списке — преимущественно старая классика, которая покажется знакомой, если вам доводилось экспериментировать с альтернативными ОС десятилетней давности или раньше: 3Com 509, NE2000, Realtek RTL8139. На том компьютере, который вы еще не засунули в чулан, должна найтись по крайней мере Intel PRO.

Пожалуй, лучше всего загрузить последнюю версию *Qemu* или *VirtualBox* из репозитория своего дистрибутива; мы больше

История древнего мира: Таненбаум против Торвальдса

Исходный код Minix 1 был опубликован на дисках и в приложении к книге Эндрю С. Таненбаума «Операционные системы: проектирование и реализация», вышедшей в 1987 г. Вскоре на Usenet образовалась группа поклонников Minix, включавшая 40 000 подписчиков. Среди них был Линус Торвальдс, который добавил в систему новые возможности. Но его расстроило нежелание Таненбаума вывести Minix за пределы обучающей системы, и в 1991 году в группе comp.os.minix Торвальдс написал: «Я пишу (бесплатную) операционную

систему (просто хобби, она не будет такой большой и профессиональной, как GNU)». Когда позже Торвальдса обвинили в краже кода Minix, Таненбаум защищал его, не без ехидства говоря: «Линус не написал код Linux внезапно и сидя в вакууме. У него была моя книга, у него был Minix, и он несомненно знал историю (она есть в моей книге). Но код был его, что доказывается тем, как он испортил архитектуру».

Спором Таненбаума–Торвальдса, описанным в приложении к книге 1999 г. «Open Sources: Голоса

революции открытого кода», в течение пары недель в начале 1992 г. была ветка на comp.os.minix, посвященная достоинствам монолитных ядер и микро-ядер, которая начинается с утверждения Таненбаума о том, что «монолитные ядра устарели».

В конце концов споры в этой ветке заглохли, но несколько лет спустя многие утверждали, что «Линус победил», основываясь чисто на рыночной доле Linux. Тем не менее, спор продолжается; последние рассуждения Таненбаума можно найти на <http://bit.ly/ReliableOS>.

экспериментировали с последним. Нажмите большую кнопку New [Создать] в левом верхнем углу окна *VirtualBox* и наберите название виртуальной машины. Мы воспользовались MINIX3. Выберите Other [Другое] в обоих выпадающих меню для типа и версии ОС; в следующем окне выделите виртуальной машине сколько памяти, сколько можете. В принципе, для Minix 3 достаточно 256 МБ, но когда вы исследуете новую систему и вам не терпится увидеть, на что она способна, чем больше памяти, тем лучше. Для одной виртуальной машины мы оставили диск размером 2 ГБ (по умолчанию), для другой — увеличили до 8 ГБ. Если вы намерены установить *все*, что может предложить Minix 3, выберите последний вариант, но помните, что в 3.3.0 не хватает некоторых программ из предыдущих релизов.

Нажмите Create [Создать], и вы увидите виртуальную машину MINIX3 в списке на левой панели. Прежде чем ее запустить, зайдите в Settings [Настройки], поставьте галочку Hardware clock in UTC time [Аппаратные часы для времени UTC] и убедитесь, что в Storage [Хранилище] указан загруженный ISO-файл. Теперь можно запускать виртуальную машину из графического интерфейса. На компьютере с процессором Core 2 Duo без расширений виртуализации VT-x/AMD-V нам пришлось запускать *VirtualBox* командой

```
VBoxSDL --startvm MINIX3 --norawr0 --norawr3
```

Далее следуйте инструкциям пошагового руководства (см. стр. 69). После окончания установки выключите компьютер командой **poweroff** вместо **shutdown**. В *VirtualBox* зайдите в Settings > Storage [Настройки > Хранилище], уберите ISO и выберите вместо него созданный образ виртуального диска. Теперь загрузите систему, нажав Start [Запуск] в *VirtualBox* (или, на компьютерах с процессорами без расширений виртуализации VT-x и AMD-V, воспользуйтесь обходным путем).

Запуск на BeagleBoard

В версии 3.3.0 версия Minix для ARM окончательно интегрирована в основную ветку кода вместе с официальной версией для x86. Нацеленная на BeagleBoard с его системой на кристалле (SoC) на базе Cortex-A8, эта версия с разной степенью успешность запускается на BeagleBoard-xM (и в ее *Qemu*-эмуляторе), BeagleBone и BeagleBone Black. Для версии 3.3.0 есть и готовые образы, но в них не будет последних дополнений. Чтобы создать образ диска для BeagleBoard, создайте каталог в `~/minix` и перейдите в него (ну или в тот каталог, где вы хотите осуществить сборку) командой **cd** и затем загрузите исходный код:

```
$ git clone git://git.minix3.org/minix minixsrc
```

Теперь перейдите в каталог **minixsrc/** и создайте файл **.settings** для BeagleBoard-xM:

```
# beagleboard-xm
U_BOOT_BIN_DIR=build/omap3_beagle/
CONSOLE=ttyO2
```

А для BeagleBone сделайте следующее:

```
#beaglebone (and black)
U_BOOT_BIN_DIR=build/am335x_evm/
CONSOLE=ttyO0
```

Если у вас установлен g++, компилятор GNU для C++, то утилиты сборки Minix обеспечат вас почти всем необходимым. В Ubuntu также нужно установить следующие пакеты:

```
apt-get install zlibc zlib1g zlib1g-dev
```

Утилиты сборки основаны на скрипте *build.sh* NetBSD. Запустите версию для ARM командой

```
./releasetools/arm_sdimage.sh
```

и вы получите образ **minix_arm_sd.img**, готовый для копирования на SD-карту для загрузки BeagleBoard или BeagleBone:

```
sudo dd if=minix_arm_sd.img of=/dev/mmcblk0 bs=1M
oflag=direct
```

Вставьте SD-карту в BeagleBoard, загрузитесь и войдите в систему от имени суперпользователя-root. Ethernet в BeagleBone не работает, а на платах BeagleBoard можно запустить **netconf** и выбрать LAN8710A. Поддержка USB в BeagleBone считается экспериментальной, и отмечены проблемы при горячем подключении USB-хабов в версии 3.3.0. Звук и АЦП тоже не работают, зато работают входы/выходы GPIO.

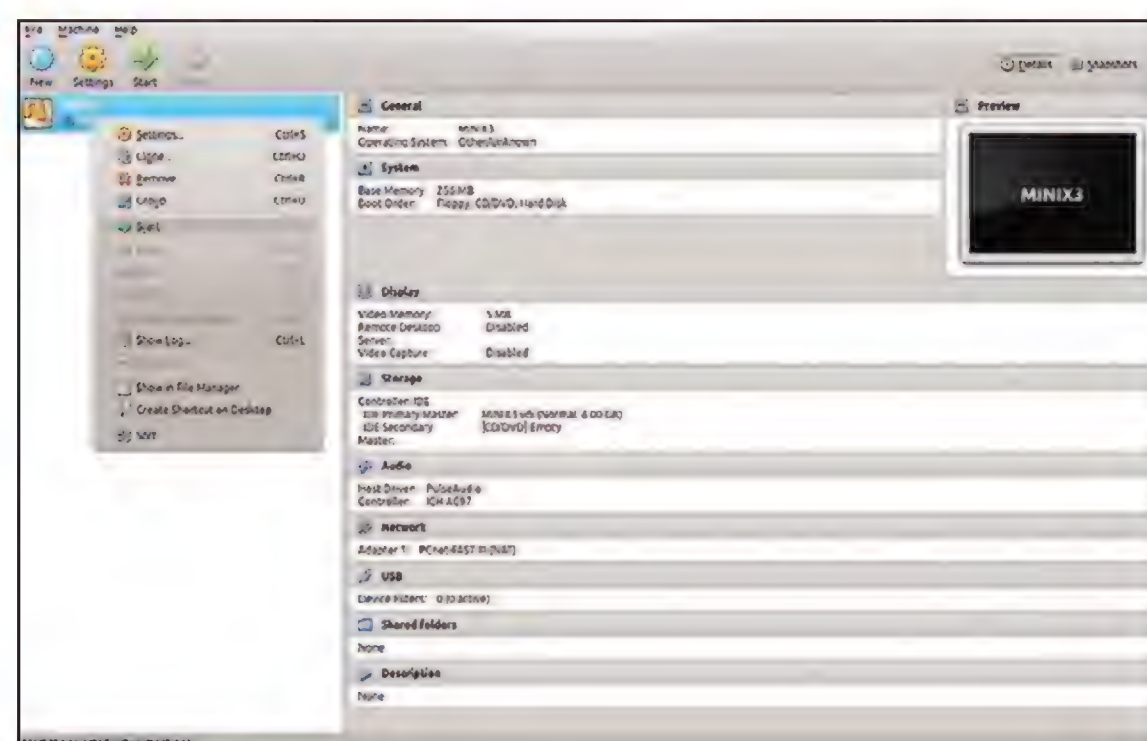
Для первого релиза это многообещающе, и BeagleBoard — хорошая «песочница» для упражнений с Minix, особенно потому, что активная работа над этим релизом будет продолжена, и вы увидите улучшения, если не перестанете следить за кодом. Также можете попробовать версию для ARM в эмуляторе *Linaro Qemu*, версии *Qemu* специально для BeagleBoard-xM. Инструкции можно найти на <http://bit.ly/LinaroQemu>.

Надежность через архитектуру

Хотя версия ARM находится на ранней стадии разработки, она вполне имеет смысл для ОС, главные достоинства которой — надежность и низкое потребление ресурсов. К встраиваемым устройствам с архитектурой ARM относятся тысячи различных



Проблемы *nix очень часто связаны либо с правами доступа, либо с DNS. Если вам не удастся заставить *pkgin* работать после установки *VirtualBox* с NAT, поместите в **/etc/resolv.conf** нашего старого друга 8.8.8.8 вместо 127.0.0.1.



» **VirtualBox** — лучший способ попробовать Minix 3, это избавит от проблем с аппаратной совместимостью.

➤ Для получения образа SD-карты для BeagleBoard используется скрипт *build.sh* NetBSD, а большинство необходимых утилит сборки загружаются на лету.

устройств, на которых задача операционной системы по сути сводится к тому, чтобы не упасть. В пользовательских устройствах среднее время работы до падения может превышать срок службы устройства, и провозглашенная Эндрю Таненбаумом цель создать «компьютеры, которым не нужна кнопка перезагрузки» выглядит здесь вполне применимой.

Своей надежностью Minix обязана модульной природе и борьбе с раздуванием. В ядре — 6000 строк (см. врезку «Ядро минималиста» внизу). Вне этих шести тысяч строк (где реализовано немногим больше, чем прерывания, планирование и обмен сообщениями) все прочие компоненты ОС (все драйверы, от консоли до жестких дисков) в Minix работают в пользовательском режиме с защитой от ошибок и атак злоумышленников. Для доступа к памяти драйверы и серверы должны просить разрешения у ядра. Ни один процесс не имеет права обращаться к ресурсам, которые напрямую ему не нужны: например, звуковой драйвер не может обращаться к диску. Принцип наименьших прав позволяет таким драйверам пользоваться только строго определенными вызовами ядра, и в соответствии с тем же принципом драйверам выделяются строгие интервалы времени, что позволяет избежать бесконечных циклов, в итоге выводящих систему из строя.

Поверх уровня драйверов располагаются управление виртуальной памятью, менеджер процессов, файловые системы и все остальное, что в традиционных системах Unix является частью ядра. Все это работает в режиме пользователя со включенным диспетчером памяти (MMU).

Уровнем выше находятся пользовательские программы, но все три уровня в пользовательском режиме — с точки зрения ядра — пользовательские процессы. В Linux падение такого кода обрушит всю систему, но в Minix 3 достаточно просто перезапустить процесс. Сервер хранения данных сохраняет состояние упавших драйверов, которое восстанавливается при запуске нового

драйвера. Мониторингом всех серверов и их перезапуском после падения, если он возможен, занимается сервер восстановления.

Это безопасно, так как большинство ошибок вызываются случайными временными ошибками и временными гонками. Команда Таненбаума проверила это, внедрив миллионы ошибок и перезаписав 100 машинных команд в запущенных драйверах Ethernet. Они внедрили 800 000 ошибок в каждый из трех различных драйверов Ethernet, вызвав 18 000 падений драйвера — и сервер восстановления каждый раз успешно реанимировал драйвер.

Конечно, вы и сами можете поатаковать ядро — исчерпывающий тест атак включен в состав системы — но сначала настроим систему. Как отмечено в пошаговом руководстве (см. стр. 69), необходимо изменить файл **repositories.conf**, указав правильный FTP-адрес Minix. Заодно раскомментируйте репозиторий NetBSD в этом файле. Если вы никогда не работали с сервером, на котором единственный текстовый редактор — это *vi*, советую установить на ноутбук *bsdgames* и играть в командной строке *Robots* до тех пор, пока клавиши H, J, K и L не закрепятся в мышечной памяти. Затем скомандуйте

pkgin update

Если при обновлении возникают проблемы, особенно с NAT на *VirtualBox*, дело скорее всего в DNS, поэтому добавьте в **/etc/resolv.conf** строку

nameserver 8.8.8.8

Теперь можно искать и устанавливать пакеты с помощью **pkgin** — например, для установки пакета *vim* выполните

```
pkgin install vim
```

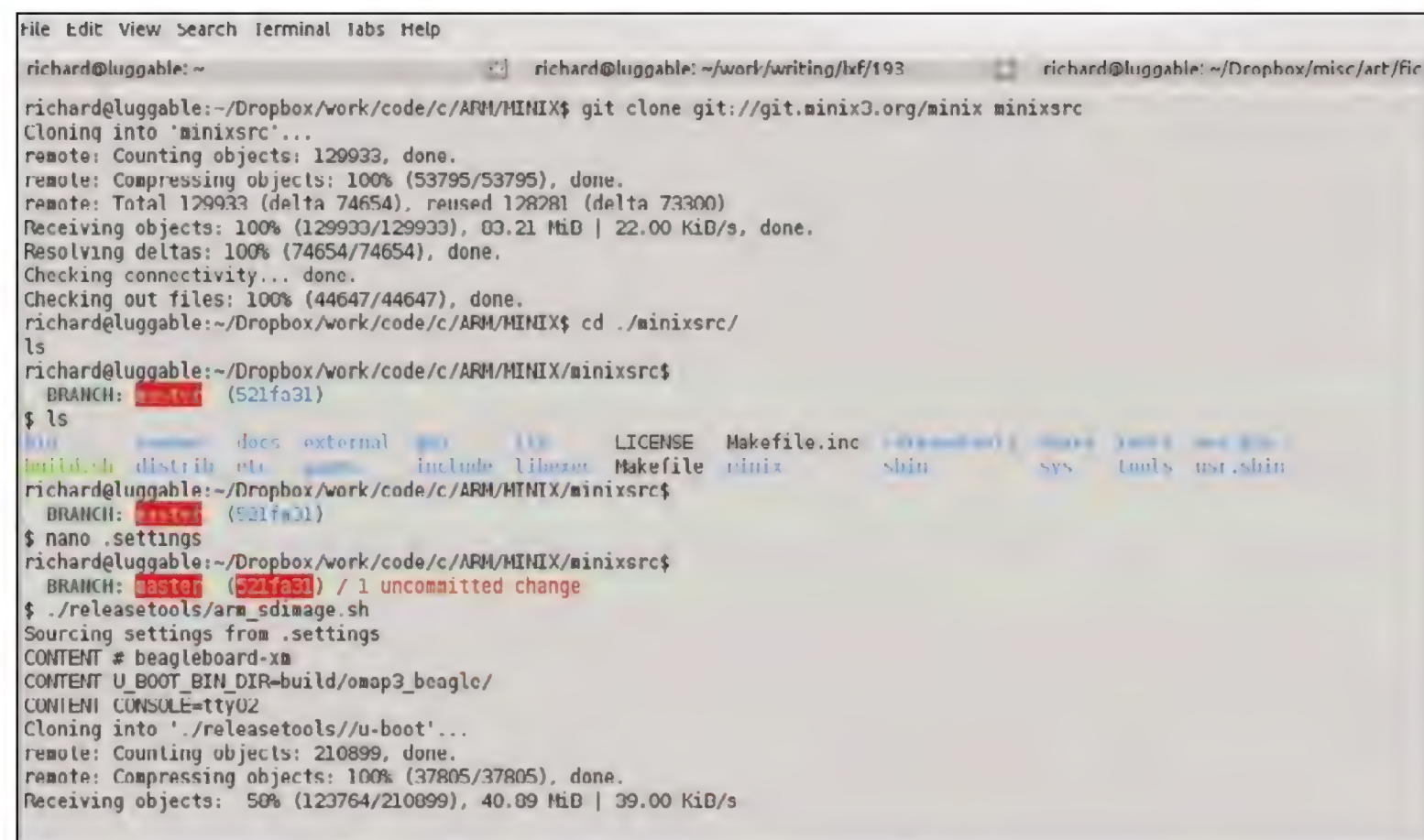
Если вы сделали раздел побольше и хотите установить сразу всё, воспользуйтесь командой

pkgin_all

Желая заглянуть поглубже, начните с вики на <http://minix3.org>, где есть, например, хорошее руководство по написанию драйверов устройств и масса другой документации для разработчиков. Пользовательская документация не то что бы плоха, но иногда отстает от актуальных релизов. Разумеется, помощь по обновлению вики приветствуется, как и помощь в программировании — на самом деле, это верхняя строчка списка пожеланий вики!

Светлое будущее микроядер?

На этом заканчивается наш обзор Minix 3 — и микроядер в целом. Возможно, нам еще далеко до удобного Hurd, но микроядра доказали свою ценность в автомобилях под управлением QnX [см. «Рулит Linux» в **LXF187**, стр. 42], маршрутизаторах Cisco и многих других сферах. Minix 3 вполне достойна внимания хотя бы как стабильная альтернатива встраиваемым системам для ARM. А с учетом ее (почти полностью объединенной) пользовательской базы NetBSD Minix становится самой удобной ОС на основе микроядра для тех, кто знаком с GNU/Linux, и мы определенно ждем улучшений в новом релизе. **LXF**



Ядро минималиста

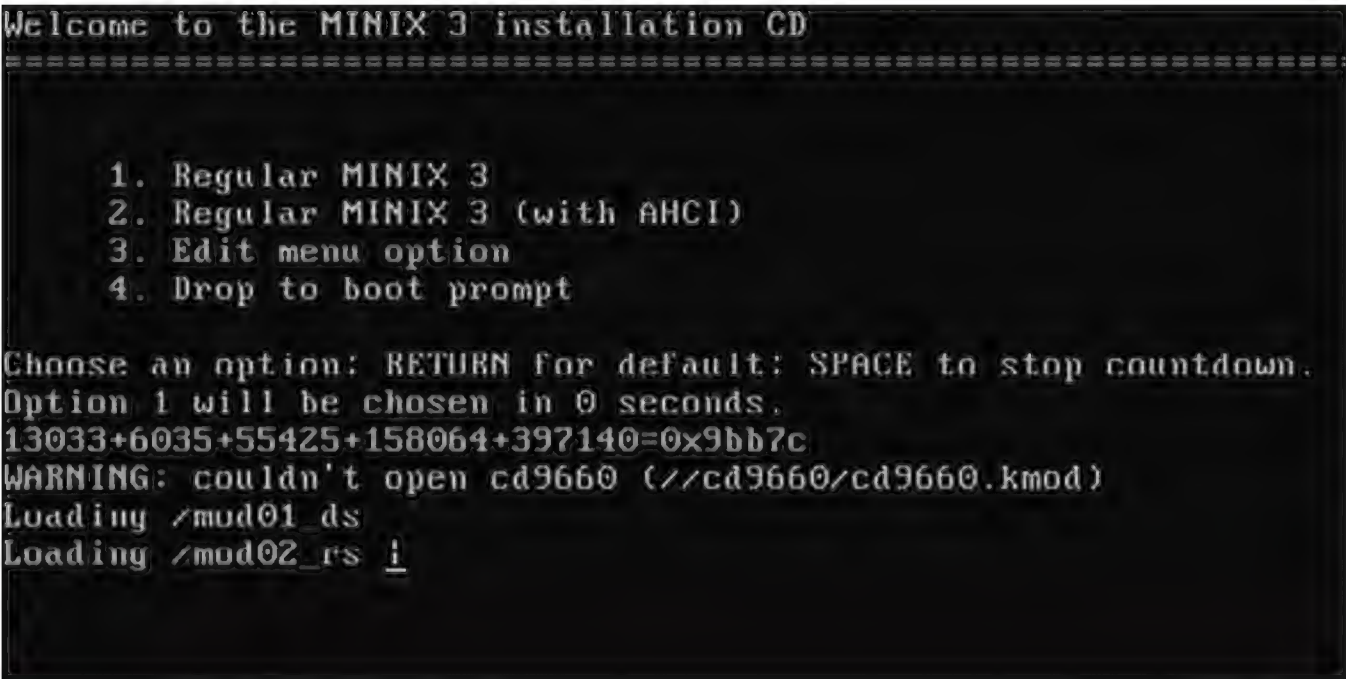
Несколько лет назад Linux преодолел рубеж в 15 миллионов строк кода. В ядре Minix 3 строк кода 6000 и примерно столько же в драйверах (традиционный код ядра, который в Minix выполняется в пользовательском режиме). Если кода в тысячу раз больше, можно предположить, что и ошибок будет в тысячу раз больше. Но все гораздо хуже.

В любом коде есть ошибки, обычно 5–10 на тысячу строк, хотя код FreeBSD получше: там их всего три на тысячу. На FOSDEM Таненбаум процитировал данные исследование Стэнфордского университета:

в коде драйвера Linux в 3–7 раз больше ошибок по сравнению с остальной частью ядра — просто потому, что копаться в непонятном коде драйвера менее увлекательно, чем в важных (и интересных) алгоритмах самого ядра. В Linux 70% кода — это драйверы. В Windows 85% падений вызывается драйверами (которые в основном пишут сторонние разработчики). Запуск этих драйверов в пространстве ядра означает, что их уязвимости могут вывести из строя всю систему; в защищенной памяти, изолированной от других процессов, область

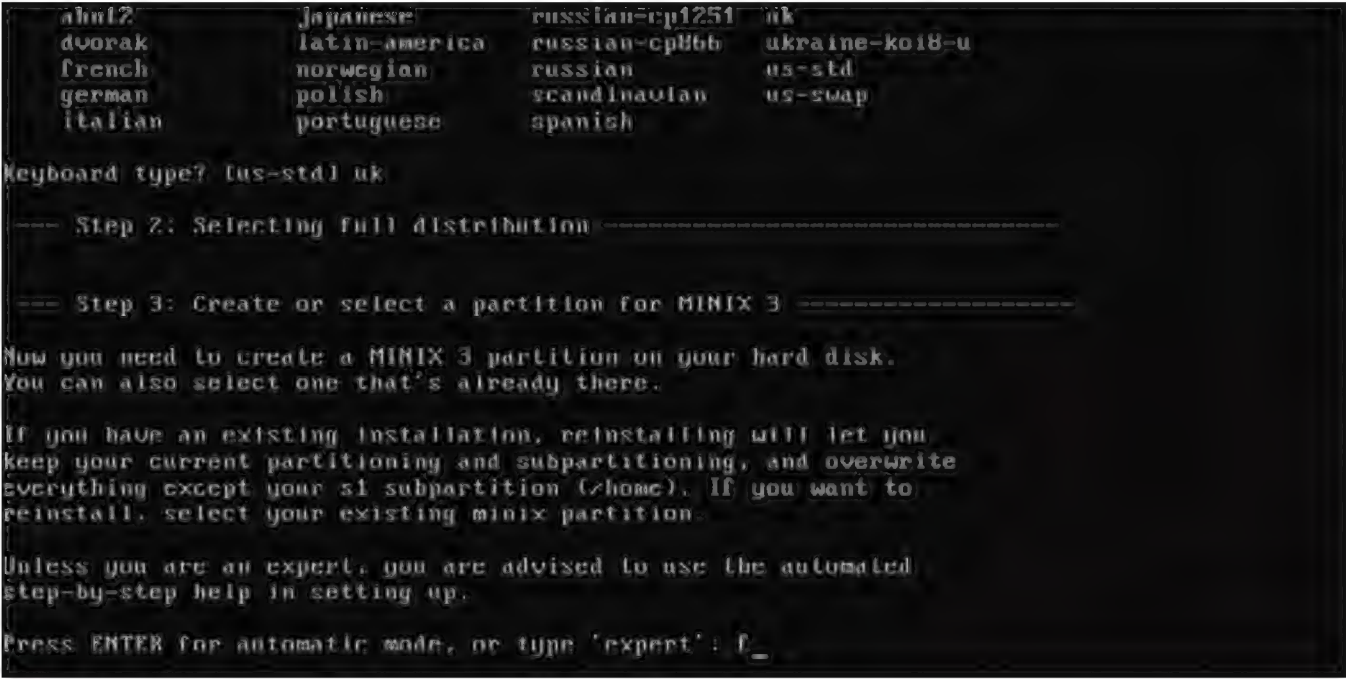
воздействия уязвимостей строго ограничена. Возьмем *Netfilter*, который заменил *ipchains* из Linux 2.2 для управления пакетами и содержит средства фильтрации пакетов, преобразования сетевых адресов и преобразования портов. В Linux *Netfilter* находится внутри ядра, а в Minix — в пространстве пользователя. Если вредоносный код снова поразит *Netfilter* (как было с уязвимостью «пакет смерти» в ядре 2.6), то в Minix будет скомпрометирован процесс пользовательского режима, а в Linux злоумышленник дорвется до власти над всей системой.

Установка Minix 3 из ISO-образа



1 Загрузка

Выберите ISO-образ (Regular MINIX 3) в меню. Игнорируйте предупреждение о cd9660 (это известная проблема, и вызывающий ее драйвер isofs отключен), все остальные модули загрузятся отлично (для виртуальных машин на более старых процессорах применяется обходной прием, приведенный в основной статье). Далее последует обычное монтирование дисков, запуск сервисов и демонов, и появится знакомая строка входа в систему.



3 UK — это не украинская?

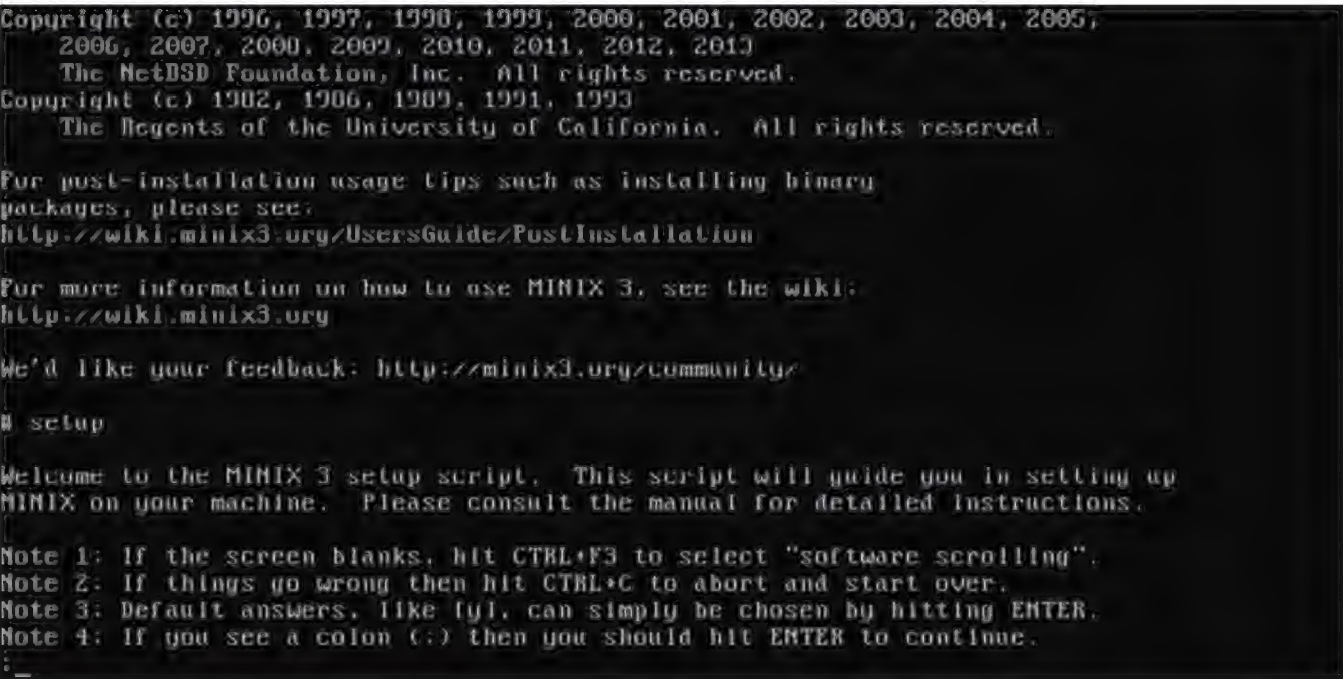
Первым делом выберите раскладку клавиатуры. В отличие от вашего дистрибутива Linux здесь по умолчанию предлагается UK, а не GB. Желая убедиться, что это не украинская раскладка, попробуйте нажать Shift+3 в следующей строке приглашения, и вы получите знак фунта. Сотрите его и нажмите Enter для автоматической разбивки диска, если у вас нет своих идей по разбивке; иначе наберите **expert**.



5 Сетевые карты

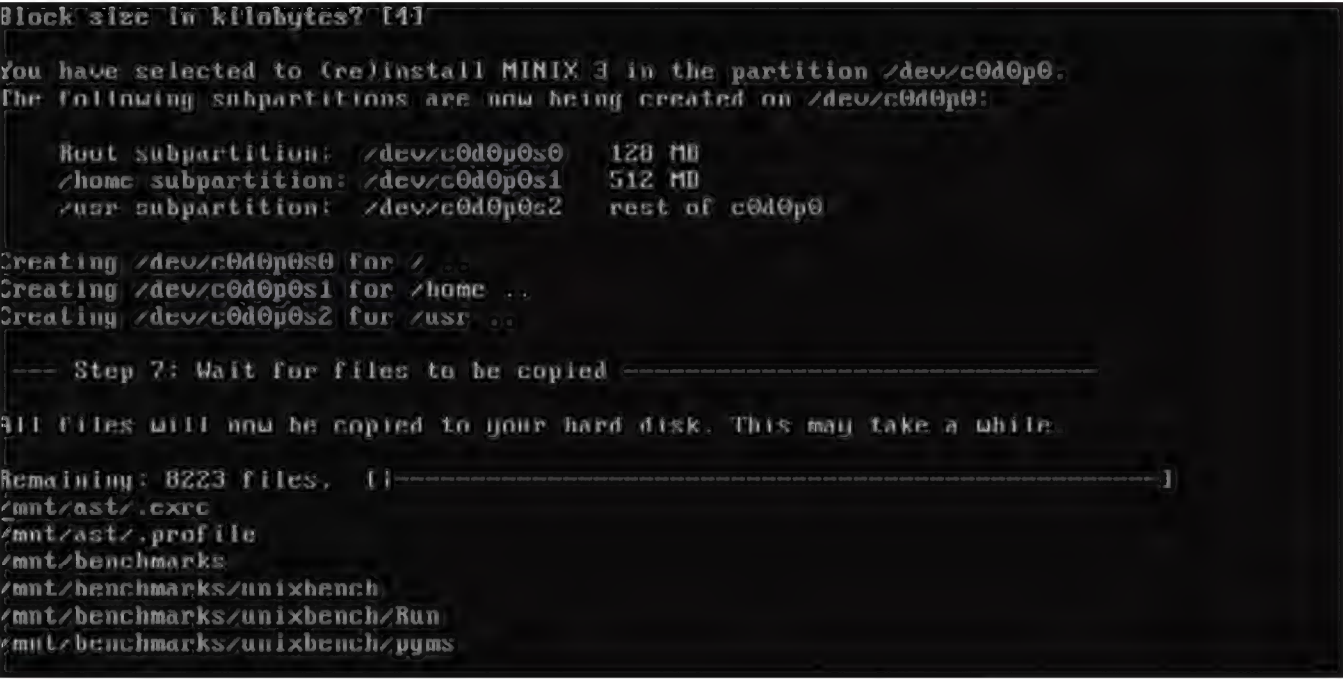
Ваша сетевая карта — если она совместима с Minix — должна определиться автоматически. Список поддерживаемых карт включает все сетевые карты виртуальных машин и несколько более старых (но хорошо задокументированных) карт. Внимательно изучите этот список, если впоследствии соберетесь устанавливать Minix 3 на настоящее «железо».

Затем нажмите Enter, чтобы принять название определенной карты (в VirtualBox и VMWare это AMD LANCE).



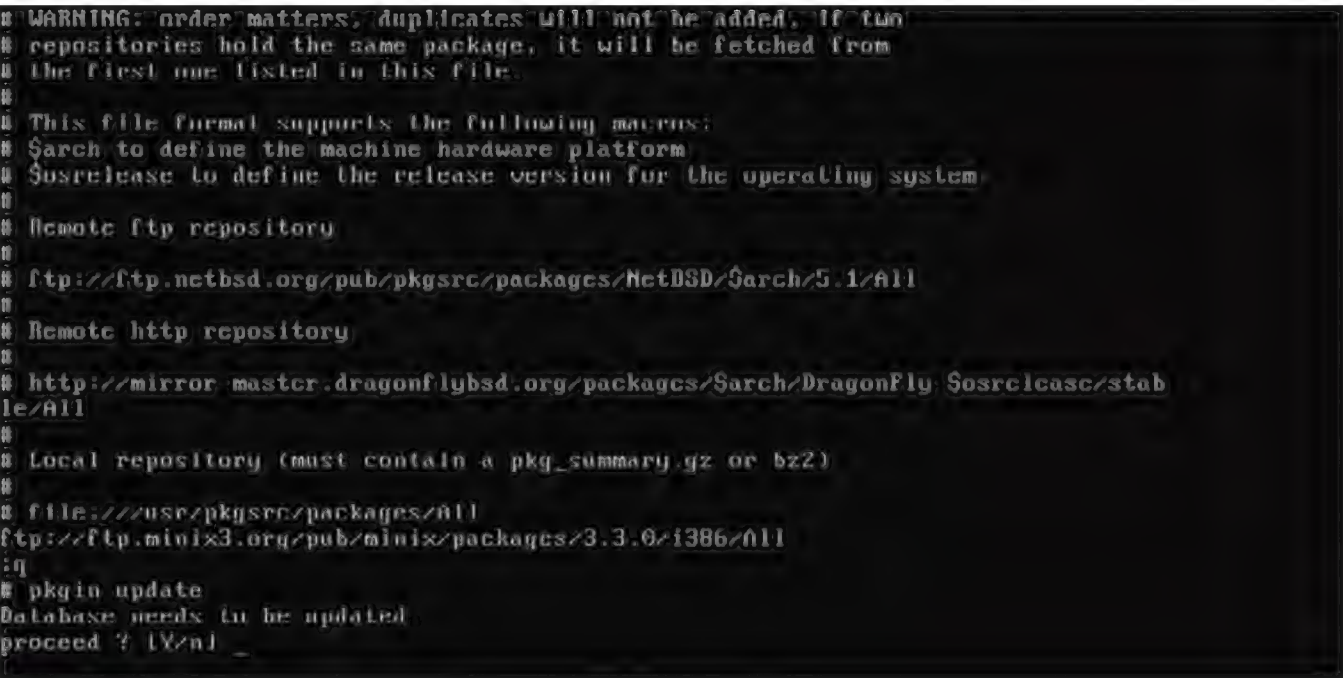
2 Встроенная документация

Зайдите в систему от имени суперпользователя-root (пока без пароля). Вы попали в среду Live-диска и можете начать исследование возможностей Minix отсюда — но если у вас запущена полноценная установка на VirtualBox, то после установки у вас появится лучшая «песочница» для Minix 3. Наберите **setup** для запуска программы установки. Прочитайте четыре строки появившегося вывода и нажмите Enter.



4 Разбиение диска

Даже при автоматической разбивке диска вам все равно придется принимать много решений, если выбран правильный регион диска (в VirtualBox это не проблема). Просто нажмите Enter, и еще раз — когда вам предложат размеры разделов и блоков. Minix отдает большую часть диска разделу **/usr**, так как здесь будет находиться большинство файлов.



6 Готово

Включив и загрузив виртуальную машину, зайдите в систему и задайте пароль root командой **passwd**. В **/usr/pkg/etc/pkgin/repositories.conf** измените строку (она должна быть единственной раскомментированной):

```
http://www.minix3.org/pub/pkgsrc/packages/$osrelease/$arch/All
```

на следующую:

```
ftp://ftp.minix3.org/pub/minix/packages/3.3.0/i386/All/
```

и, наконец, запустите **pkgin update**.

HDD: Стереть диски навсегда

Шон Конвэй обобщает методы надежного удаления данных со старых жёстких дисков — от *DBAN* и *Secure Erase* до стрельбы на поражение!



Наш эксперт

Шон Конвэй — вечный студент с опытом работы с электронными лампами и комплектными металло-оксидными полупроводниками.

Вы подумываете о пожертвовании или продаже пары подержанных компьютеров. И вы осознаёте важность очистки ваших данных для защиты конфиденциальности — так что же вам делать? По словам Национального института стандартов и технологий, есть три степени очистки: поверхностная, глубокая и полное разрушение.

Поверхностная очистка предотвращает восстановление данных утилитами восстановления файлов. Глубокая очистка не позволит технарям применить для восстановления данных оборудование обработки сигналов, а разрушение — это... ну, разрушение физического носителя, до полной невозможности использовать повторно.

Для нашего урока мы протестировали на жёстком диске ноутбука (160-ГБ 5400.6 Seagate) семь разных методов очистки: стандартные команды **rm**, **format** и **shred**; ПО *DBAN* и прошивки *Secure Erase*; размагничивание в специальной машине; и физическое разрушение. Не каждый читатель имеет доступ к оборудованию стоимостью \$10 000 или желание рискнуть жёстким диском исключительно для того, чтобы полюбоваться последствиями, так что вы, дорогие читатели, извлекаете выгоду без малейшей опасности для своих устройств.

Вы можете прочитать больше о проведении наших тестов на соседней странице (см. «Наша процедура тестирования»). Результаты сведены в следующую таблицу:

Метод	Тип очистки	Результат восстановления
Команда rm	Ниже минимума	Читаемо: 3 файла
Команда format	Ниже минимума	Читаемо: 3 файла
Команда shred	Очистка: только конкретных файлов	Читаемо: 0 файлов
<i>DBAN</i>	Очистка: максимум времени	Читаемо: 0 файлов
<i>Secure Erase</i>	Очистка: минимум времени	Читаемо: 0 файлов
Размагничивание	Очистка: спецоборудование	Читаемо: 0 файлов
Разрушение	Разрушение	Читаемо: 0 файлов

Как видите, пять из этих методов обеспечивают некую степень очистки. И выбор определённого метода зависит от конкретной ситуации. Скажем, команда **shred** делает нечитаемыми файлы данных, но запускается вручную для каждого файла, а потому неудобна для очистки диска целиком. А вот размагничивание и разрушение делают жёсткий диск нечитаемым, так что не применяйте эти методы, если хотите, чтобы диск пригодился новому владельцу.

На уроке мы покажем, как внедрить в жизнь последние четыре метода из таблицы. (Подозреваем, что с командой **shred** вы и сами справитесь.)

Предупреждаем: не пробуйте исследовать описанные в нашем уроке методы на своем личном компьютере. Инструменты, которые мы применяем, могут сделать жёсткий диск нечитаемым. Если вы намерены следовать уроку, используйте отдельный компьютер с таким жёстким диском, который не жалко пожертвовать во имя просвещения.

Данные удаляет DBAN

DBAN (Darik's Boot And Nuke, <http://dban.org>) — это загружаемое приложение, распространяемое Blancco. Бесплатный образ CD, предоставляемый изготовителем, разработан для очистки жёсткого диска в домашних условиях, но компания также имеет лицензионную коммерческую версию для бизнес-приложений. Это подборка алгоритмов очистки и настроек, которые пользователь выбирает для удаления содержимого целевого жёсткого диска.

После запуска программа перезапишет жёсткий диск, заполнив его случайными или заранее заданными данными. Процесс переписывает разделы жёсткого диска с данными файлов и данными системных файлов, плюс все выбранные места на жёстком диске. Целью здесь является замена данных на жёстком диске на случайные данные, сгенерированные алгоритмом. *DBAN* может применяться только на исправных жёстких дисках: тех, что обнаруживает BIOS, и пригодных для работы.

Скачайте *DBAN 2.2.8* с сайта и создайте загрузочный CD/DVD. Когда программа загрузится, выберите из списка *DBAN*, а затем — любой из обнаруженных жёстких дисков. Изменения в программу вносятся с помощью сочетаний клавиш, перечисленных в нижней части экрана.

Выберите диск, который хотите очистить, и нажмите F10 для начала очистки. Раздел статистики появится на экране сразу после начала процесса. Это может занять немало времени, поскольку *DBAN* приходится генерировать и писать данные для каждого бита памяти на жёстком диске.

Применение Secure Erase

В порядке альтернативы, жёсткие диски, произведённые после 2001 г., идут с прошивками, которые обеспечивают очистку, не вовлекая дополнительные программы. Жёсткий диск ATA, определяемый системным BIOS, можно очистить, введя в окне терминала команду *Secure Erase* [надежная очистка]. Любая введённая команда выполнится прямо на жёстком диске.

В течение этого процесса жёсткий диск может заблокироваться и стать недоступным для ОС. При тестировании мы умудрились

Скорая помощь



Если вы не прочь рискнуть, то в Fedora и Ubuntu имеется набор инструментов *hdparm*, который можно установить с помощью, соответственно, *yum* и менеджера пакетов *apt-get*.

Наша процедура проверки

Прежде чем приступить к каждому из методов очистки жёсткого диска, мы провели диск через стандартную процедуру подготовки. Вначале мы очистили его от данных, используя ПО для удаления данных с CD *DBAN* — RCMP TSSIT OPS-II. Канадское правительство объявило этот метод устаревшим и заменило на CSEC ITSG-06 — метод, использующий преимущества Secure Erase. Метод RCMP мы выбрали не потому, что он лучше других, а потому что для автора-канадца в нём есть особый шарм Red Serge [красный цвет формы канадской полиции, — прим. пер.].

Очистив данные, мы разбили жёсткий диск на разделы (с помощью **fdisk**), отформатировали в ext3 (с помощью **mkfs.ext3**) и скопировали на него девять разных типов файлов (DOC, DOCX, EPUB, JPG, PNG, ODS, ODT, TXT and ZIP), подготавливая к проверке.

По отработке каждого из методов очистки мы использовали приложение для восстановления файлов *PhotoRec*, описанное в позапрошлом номере (см. Учебники **LXF190**, стр. 62), чтобы попытаться восстановить файлы.

```
sconway@fedora19-inane-ca ~]$ sudo fdisk -l /dev/sdc
Disk /dev/sdc: 160.0 GB, 160041885696 bytes, 312581808 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x312121e6

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1             2048     312581807     156289880   c3   Linux
[sconway@fedora19-inane-ca ~]$ sudo mkfs.ext3 /dev/sdc1
mke2fs 1.42.7 (21-Jan-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
9773056 inodes, 39072470 blocks
1953623 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
1193 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
```

» О Канада! Наш дом и родная земля! Сынам твоим без Secure Erase нельзя! [аллюзия на государственный гимн Канады, — прим. пер.]

ненамеренно сделать это, сделав жёсткий диск бесполезным. (К счастью, мы сумели его восстановить, применив определяемую пользователем команду пароля безопасности).

По этой причине риск, связанный с лёгкостью доступа к Secure Erase, не мешает снизить, взяв отдельную ОС специально для тестирования: нечто вроде загрузаемого SystemRescueCd (www.sysresccd.org), который содержит набор команд Secure Erase.

Образ SystemRescueCd — это набор инструментов для администрирования или восстановления компьютеров с Linux или Windows. Программа загружается с CD/DVD или USB, не требуя установки, и меню выбора при загрузке позволяет пользователю работать в CLI или GUI.

На сайте достаточно документации, поясняющей, как создать загрузочный CD, DVD или USB. Загрузив SystemRescueCd с удобного для вас носителя, вы сможете выбрать поддержку для интерфейса GUI в первом меню. Вы получите доступ root к полной ОС Linux и поддерживаемым приложениям. Если система подключена к Интернету, часть загрузочного процесса потребует IP-адрес через DHCP, чтобы установить доступ к сети и, как мы надемся, к Интернету.

В коде к этому уроку вы увидите два параметра: **X** и **#**. **X** — это ID устройства. Эта информация может быть получена из приложения вроде *GParted* или вытащена из лога *dmesg*. **#** — номер хоста. Мы определим его позднее.

Откройте терминал. Сейчас вы будете использовать команду **hdparm** (см. «Скорая помощь», стр. 70) следующим образом:

```
hdparm -l /dev/sdX
```

В результате вы должны увидеть нечто вроде

```
Security:
Master password revision code = 65534
supported
not enabled
not locked
frozen
not expired: security count
supported: enhanced erase
```

Чтобы продолжить с Secure Erase, у жёсткого диска должны быть установлены параметры **not frozen** [не «заморожен»], **not locked** [не заблокирован] и **enabled** [включён]. Wiki (<http://bit.ly/SecureErase>) даёт объяснение, почему могут возникать эти условия и как их исправить.

В наших проверках мы преуспели в установке параметра **not frozen**, применив способ, описанный Эдоардо Ливерани [Edoardo Liverani] (<http://bit.ly/Liverani>), и сейчас о нём расскажем.

Примечание: жёсткие диски, использованные нами во время проверок, поддерживают режим «горячей замены» (то есть их можно отсоединять, не прекращая работы компьютера). Но отключать обычный жёсткий диск или устанавливать компоненты при работающей системе — очень плохая затея.

Прежде всего надо определить номер хоста:

```
ls -ld /sys/block/sdX
```

В выводе этой команды взгляните на текстовую строку вроде **host5**. Это и есть нужный вам номер хоста. Отсоедините жёсткий диск и выключите компьютер. (Помните о том, что мы сказали о запрете «горячей замены» для обычных жёстких дисков.) Теперь запустите

```
echo 1 > /sys/block/sdX/device/delete
```

Присоедините жёсткий диск и запустите

```
echo «-- --» > /sys/class/scsi_host/host#/scan
```

Проверьте состояние жёсткого диска с помощью **hdparm** и убедитесь, что параметр **frozen** установлен.

Чтобы продолжить работу с Secure Erase, следующий параметр безопасности должен быть изменён на **enabled**. Чтобы

```
Darik's Boot and Nuke 2.2.8

Options:
Entropy: Linux Kernel (random)
PRNG: Mersenne Twister (mt19937ar-cok)
Method: RCMP TSSIT OPS-II
Verify: Last Pass
Bounds: 1

Statistics:
Runtime:
Remaining:
Load Averages:
Throughput:
Errors:

Disks and Partitions:
[ ] SCSI Disk USB2.0 Flash Disk 1100 3023MB
[ ] SCSI Disk Seagate FA Goflex Desk 0157 1397GB NA0LEJF4
```

» Высокотехнологичная очистка диска с помощью *DBAN*, записанная на куда менее высокотехнологичную камеру.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe!

Скорая помощь

Вся документация Secure Erase предупреждает о риске. Некоторые из команд даже требуют атрибута `--i-knowwhat-i-am-doing` [я отдаю себе отчет в своих действиях] перед выполнением.

включить безопасность на жёстком диске, введите следующее в командной строке:

```
hdparm --user-master u --security-set-pass sean /dev/sdX
```

Вместо **sean** можете подставить любой удобный вам пароль. Желая вернуть параметр безопасности жёсткого диска к **not enabled**, скомандуйте

```
hdparm --user-master u --security-disable sean /dev/sdX
```

Для удаления опции безопасности **locked** введите

```
hdparm --user-master u --security-unlock sean /dev/sdX
```

Запускается Secure Erase с помощью

```
hdparm --user-master u --security-erase sean /dev/sdX
```

Прежде чем вернуться к работе с жёстким диском, убедитесь, что параметры **not enabled**, **not locked** и **not frozen** снова установлены, иначе ОС не получит доступа к жёсткому диску.

Размагнитим и уничтожим

В завершение нашей дискуссии рассмотрим два последних метода очистки. Если поместить жесткий диск в размагничиватель или физически уничтожить его, раздробив или разбив на куски, то, вполне очевидно, и данные, и сам диск уже будут непригодны для дальнейшего использования. Но как эти методы работают в реальности?

Жёсткий диск состоит из круглых металлических или стеклянных пластин, покрытых окисью железа. Каждая пластина

вращается между металлическими катушками, которые называются пишущими головками. Когда пишущая головка получает электрические сигналы, она примагничивает поверхность к пластине. Данные записаны на пластине в виде ряда магнитных единиц и нулей.

Размагничивающее устройство для жёстких дисков — это электронный прибор, генерирующий мощное магнитное поле. Поле сотрёт данные, и их нельзя будет извлечь. Помещение жёсткого диска в магнитное поле размагничивателя удаляет магнитную информацию на жёстком диске.

Для очистки нашего тестового жёсткого диска мы использовали размагничивающее устройство Garner HD-3WXL. (Вы можете увидеть видео этого процесса на <http://bit.ly/Degausser>). Принцип его работы основан на разрядке конденсаторов. Внутренняя электроника заряжает конденсаторы для хранения энергии, аналогично зарядке батареи. Когда конденсаторы в устройстве разряжаются через специальные катушки, создаётся мощный электромагнитный импульс (ЕМР — электромагнитный импульс, ЭМИ). Магнитное поле столь сильно, что оно перемешивает все магнитные материалы в жёстком диске. ЭМИ, созданный размагничивающим устройством, мало чем отличается от импульса при ядерном взрыве.

Размагничивание уничтожает не только все данные пользователя, но также вспомогательную информацию о дорожках, проложенных на пластинах жёсткого диска при изготовлении. Без этой информации жёсткий диск больше не будет работать, так что размагниченный жёсткий диск не загрузится.

Размагнитив жёсткий диск, мы установили его на наш тестовый компьютер и попытались запустить его. BIOS обнаружил жёсткий диск, но не смог проанализировать его, доложив об ошибке в жёстком диске или его прошивке, и только потом продолжил цикл загрузки.

Крайние меры

Нашим последним способом очистки было физическое уничтожение жёсткого диска. Это означает разбить его пластины на кусочки, чтобы сделать его чтение невозможным. Это можно сделать специализированным шредером для металла или механической дробилкой. У Security Engineered Machinery, производителя подобного оборудования, есть немало клёвых видео про уничтожение жёсткого диска: <http://www.semshred.com/videos>.

Если у вас нет шредера, можно продырявить пластины мощной дрелью или поцарапать их поверхность. Есть мнение, что достаточно просто погнуть пластину, чтобы предотвратить чтение данных. Однако некоторые теоретики предполагают, что у правительственных организаций есть технологии, позволяющие восстановить данные, если пластины не уничтожены полностью.

И, наконец, автор воспользовался методом разрушения, применяя который не рекомендуется, особенно если вы живёте в Великобритании. Но в Канаде с этим не такая большая проблема.

На диске этого месяца есть видео про уничтожение тестового жесткого диска пулей из магазинной винтовки Ли-Энфилда со скользящим затвором. Мы бы употребили и бейсбольную битку, воссоздав сцену из «Офисного пространства» [американская комедия, — прим. пер.], но не были убеждены, что бита нанесёт достаточный урон.

Итак, системные администраторы, вот вам ряд методов очистки, обеспечивающих отсутствие персональных данных на жёстких дисках, которые вы хотите отдать. Надеемся, вы найдёте себе метод, отвечающий вашим требованиям, и избавитесь от этих старых компьютеров. **LXF**



» Терминатор мира очистки данных: HD3WXL Data Eliminator.

DBAN и UNetbootin

Если в дистрибутиве Linux для создания загрузочного USB вы предпочитаете *UNetbootin*, а не образ *DBAN*, понадобится немного отредактировать файлы, чтобы USB работал. По умолчанию загрузочный USB, созданный при помощи *UNetbootin*, тормозит и выдаёт сообщение об ошибке RAM-диска, что и неплохо, поскольку настройки, созданные *UNetbootin*, во время загрузки сметают данные на жёстком диске командой **autonuke**, не спросив у пользователя.

Алекс Паундс [Alex Pounds] предоставил в своём блоге (<http://bit.ly/UNetbootinBugs>)

обходной путь, который мы перефразировали для вас ниже.

В **syslinux.cfg** на USB-носителе, созданном с помощью *UNetbootin*, выполните следующие изменения в тексте этого файла:

» Замените все вхождения **ubninit** на **ISOLINUX.BIN**.

» Замените все **ubnkern** на **DBAN.BZI**.

» Удалите **--autonuke** из строки в **label unetbootindefault**.

Не забудьте сохранить изменения — и помните, что записи чувствительны к регистру.

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

ДИСТРИБУТИВЫ GNU/LINUX НА USB FLASH



Linux Mint 16



Ubuntu 14.04

А ТАКЖЕ

версии для юриди-
ческих лиц —
с лицензионным
договором
присоединения

USB Flash 8 Gb

495 рублей

www.linuxcenter.ru/shop/distros/usb/

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center

ipset: Упростим брандмауэр

Даниил Батурин покажет, как снизить сложность управления брандмауэром, применяя *ipset* для создания легко читаемых и модифицируемых конфигураций.



Наш эксперт

Даниил Батурин — программист и сетевой администратор, а также куратор дистрибутива VyOS для роутеров (<http://vyos.net>) и ещё нескольких свободных программных проектов.



Компьютерные сети становятся всё сложнее, а люди полагаются на них гораздо больше, чем когда-либо. Брандмауэры по-прежнему остаются первой линией защиты, а это означает, что сложность политик брандмауэров также растёт.

Иногда системным администраторам поступают просьбы наподобие «разрешить HTTP для узла 192.0.2.1», но чаще эти запросы более обобщённые, например, «разрешить SSH со всех машин разработчиков», «разрешить HTTP и HTTPS со всех офисных компьютеров» или «разрешить SMTP, IMAP и что-то ещё для этих сайтов».

Также вам может понадобиться разрешить или запретить сервисы, требующие больше одного протокола и порта, например, IPsec, который использует пару IP-протоколов для данных и UDP для обмена ключами; или SIP, умеющий применять для соединения как TCP, так и UDP, и требующий набор портов UDP для медиа; или даже Active Directory — тут нужна почти дюжина портов TCP и UDP.

Конечно же, вы можете сделать всё это, используя только *iptables*. Очевидная проблема в том, что для выполнения сложных требований часто нужно сразу несколько правил. Иногда вы можете совладать со сложностью протокола при помощи состояния RELATED и использовать модули *conntrack*; иногда вы можете охватить несколько адресов и портов написанием своих скриптов или использованием уже готовых, вроде инструмента *Shorewall* для конфигурации. Но скрипты, которые генерируют правила *iptables*, не уменьшают сложность, а лишь прячут её, так что вывод **iptables -L** остаётся не меньшим, чем если бы вы писали правила вручную (а иногда и больше). Если вы считаете, что сложность правил становится проблемой, одно из её решений — *ipset*.

Схема работы

Обычно работа с *ipset* выглядит так: вы создаёте список [set], добавляете в него несколько записей и создаёте правило для *iptables/ip6tables*, которое на него ссылается.

Звучит-то просто. Списки могут быть разных типов, и нельзя добавлять записи в список, если этот тип не указан при создании. Нельзя ссылаться на список из правила *iptables*, пока он не создан. Хорошая новость в том, что список, уже используемый *iptables*, можно изменять на лету, не перезагружая ни одно из правил. Итак, предположим, что у нас есть следующие требования:

» Разрешить SMTP, IMAP и POP3 из доверенных сетей 192.0.2.0/24 и 2001:db8::/64



Скорая помощь

Для списков IPv4-адресов используйте опцию **family inet**, а для списков IPv6-адресов — опцию **family inet6**. Если семейство адресов не указано, предполагается IPv4.

Ipset — это расширение для *Netfilter*, позволяющее создавать списки адресов IPv4 и IPv6, а также сетей и портов TCP/UDP, и употреблять эти списки для полей source [адрес отправителя] и destination [адрес получателя] в настройках *iptables/ip6tables*. В сложных конфигурациях брандмауэров это упрощает чтение и изменение настроек. Если ваша конфигурация содержит много почти идентичных правил с небольшими вариациями адресов или портов отправителей/получателей, то *ipset* — для вас.

Ipset состоит из двух частей: модуля ядра и инструмента пространства пользователя для администрирования. Первая часть сейчас входит в основную версию ядра, а инструмент обычно доступен в репозиториях. Некоторые дистрибутивы имеют сервисы-обёртки для загрузки конфигураций *ipset* при включении, например, *ipset-service* в Fedora.

» Разрешить SSH с узлов 192.0.2.10, 192.0.2.15, 2001:db8::100, 2001:db8::105

» Разрешить соединения с порта 5000 узла 203.0.113.5, порта 5010 узла 203.0.113.10 и порта 5020 узла 203.0.113.42

Посмотрим, как сократить количество правил и сделать настройку легче в обслуживании. Первым делом разберёмся с почтовой частью. Как видите, портов здесь больше, чем сетевых адресов, вот и начнём с того, что создадим список портов и сошлёмся на него в правилах для этих подсетей:

```
# ipset create EmailPorts bitmap:port range 0-65535 comment
# ipset add EmailPorts 25 comment SMTP
# ipset add EmailPorts 110 comment POP3
# ipset add EmailPorts 143 comment IMAP
```

В команде **create** мы указали имя списка — **EmailPorts**. Тип **bitmap:port** — это тип, необходимый для хранения портов. Параметр **range** (диапазон) обязателен для **port**, но вы, если хотите, можете сузить диапазон. Расширение *ipset* для хранения комментариев вместе с записями опционально, поэтому его следует явно включить параметром **comment**.

Теперь вы можете убедиться, что всё правильно, посмотрев только что созданный список следующей командой:

```
ipset list EmailPorts
```

Также можно просмотреть все созданные списки с помощью **ipset list** без аргументов. Теперь, имея список, мы можем сослаться на него в правиле *Netfilter*:

```
# iptables -A INPUT -s 192.0.2.0/24 -p tcp -m set --match-set
EmailPorts dst -j ACCEPT
# ip6tables -A INPUT -s 2001:db8::/64 -p tcp -m set --match-set
EmailPorts dst -j ACCEPT
```

Заметьте, что напрямую указать протокол вместе с портом в этом типе списка нельзя; в правиле нужно указывать именно протокол. Это работает одинаково для TCP и UDP (и даже SCTP). Опцией **-m set --match-set** вы ссылаетесь на свой объект *ipset*. Аргумент этой опции состоит из двух частей: списка и направления [direction]. Насчет списка все очевидно: это просто имя вашего списка. Направление должно быть **src** (source, отправитель) или **dst** (destination, получатель — далее мы увидим, что направлений

```
File Edit Tabs Help
[root@eirin ~]#
[root@eirin ~]# ipset create EmailPorts bitmap:port range 0-65535 comment
[root@eirin ~]# ipset add EmailPorts 25 comment SMTP
[root@eirin ~]# ipset add EmailPorts 110 comment POP3
[root@eirin ~]# ipset add EmailPorts 143 comment IMAP
[root@eirin ~]#
[root@eirin ~]# ipset list
Name: EmailPorts
Type: bitmap:port
Revision: 2
Header: range 0-65535 comment
Size in memory: 532596
References: 0
Members:
25 comment "SMTP"
110 comment "POP3"
143 comment "IMAP"
[root@eirin ~]# █
```

может быть больше одного). В данном случае нас интересуют порты у получателя, поэтому мы использовали **dst**.

Списки портов могут использоваться как в *iptables*, так и в *ip6tables*; для любого из протоколов нет необходимости что-либо править. Если вам нужно добавить к списку целый диапазон портов, существует сокращённый вызов для этого:

```
# ipset add PortList 15000-16000
```

Оборотная сторона именно в том, что это сокращённый вызов, и на самом деле *ipset* добавит в список все порты из диапазона, что может серьёзно снизить читаемость, поэтому для чересчур больших диапазонов может быть лучше указывать их в правилах *iptables/ip6tables*.

Список узлов

Перейдём к требованию об SSH. В данном случае у нас есть несколько узлов и всего один порт, поэтому есть смысл создать список узлов и затем сослаться на него в правиле для SSH. В *ipset* нельзя смешивать IPv4 и IPv6 в одном списке — так же, как

» Неплохо назначать портам запоминающиеся имена, чтобы вы и через месяц вспомнили, для чего предназначен каждый из них.

IPv6 против IPv4

По части перенаправлений, фильтрации и политик IPv6 не так уж сильно отличается от IPv4. Наиболее заметная разница — в том, что команда **iptables** называется **ip6tables**, но все опции, кроме некоторых действительно специфичных для протокола, те же самые. Аргументы для *ipset* — не исключение: вы можете использовать один и тот же формат для *iptables* и *ip6tables* без необходимости запоминать какие-то хитрости.

У других опций есть двойники, названные в IPv6 по-другому: время жизни в IPv4 называется TTL (**-m ttl --ttl=[eq|lt|gt]**), а в IPv6 — HL (hop limit, «верхний предел»). Эквивалентом

в IPv6 будет **-m hl --hl=[eq|lt|gt]**. Другой пример — протокол ICMP, который для IPv6 немного отличается. Параметр для сообщений ICMP: **-m icmpv6 --icmpv6-type=<тип>**.

Конечно, у некоторых аргументов нет аналогов в IPv6. К таким аргументам относятся заголовок для мобильного IPv6 [Mobility Header], параметры для получателя [Destination Options], параметры для каждого узла [Hop-by-Hop Options] и ещё несколько. Однако они не используются так широко, как опции вроде адреса отправителя и получателя. Все ваши знания об *iptables* будут отнюдь не лишними и при переходе на IPv6.



» Интернет постепенно переключается на IPv6, и пора изучать работу с этим протоколом. К счастью, многие опции похожи на IPv4.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe!

вам приходится использовать для разных протоколов *iptables* и *ip6tables*. Поэтому мы сделаем два списка:

```
# ipset create TrustedHosts hash:ip family inet comment
# ipset add SSH_Hosts 192.0.2.10 comment "Alice's workstation"
# ipset add SSH_Hosts 192.0.2.15 comment "Bob's workstation"
```

Здесь **hash:ip** — это тип для хранения адресов узлов IPv4 либо IPv6, а **family inet** уточняет семейство адресов IP. Если семейство не указано, считается, что это IPv4, так что в данном случае опция избыточна. Заметьте, что она обязательна для IPv6:

```
# ipset create TrustedHosts6 hash:ip family inet6 comment
# ipset add TrustedHosts6 2001:db8::100 comment «У Алисы
рабочая станция IPv6»
# ipset add TrustedHosts6 2001:db8::105 comment « У Боба
рабочая станция IPv6»
```

```
Осталось только настроить правила Netfilter:
# iptables -A INPUT -p tcp --dport 22 -m set --match-set
TrustedHosts src -j ACCEPT
# ip6tables -A INPUT -p tcp --dport 22 -m set --match-set
TrustedHosts6 src -j ACCEPT
```

В одном правиле допускается использовать несколько опций **--match-set**. Если бы нам нужно было также разрешить электронную почту с этих узлов, мы могли бы повторно использовать созданный ранее список **EmailPorts** и сделать нечто вроде

```
# iptables -A INPUT -p tcp -m set --match-set EmailPorts dst -m
set --match-set TrustedHosts src -j ACCEPT
```

Существует также тип для списка сетей, **hash:net**, который хранит адреса подсети:

```
# ipset create NetworkList hash:net
# ipset add NetworkList 10.1.0.0/24
```

IP-адреса и порты

Теперь мы поработаем с более сложным случаем: список не отдельных объектов, а пар. В нашем примере это список пар IP-адресов и номеров портов, каждый из которых требует нескольких портов и адресов. Пример может показаться немного надуманным, но я встречал ситуации, когда на первый взгляд случайные

адреса и порты были на самом деле связаны друг с другом: например, нужно было выдать инженерам поддержки доступ к приложению, работающему на внутренней машине за NAT, на предприятии с несколькими офисами.

Ipset поддерживает пары (и даже тройки) узлов и портов или сетей и портой. Тип для пар «адрес и порт» — **hash:ip,port**:

```
# ipset create AppSupport hash:ip,port
# ipset add AppSupport 203.0.113.5,tcp:5000
# ipset add AppSupport 203.0.113.10,tcp:5010
# ipset add AppSupport 203.0.113.42,tcp:5020
```

В данном случае в опции **--match-set** нужно указывать два направления — первое для адресов, второе для портов:

```
# iptables -A INPUT -m set --match-set AppSupport src,dst -j
ACCEPT
```

Можно указать любую комбинацию **src** и **dst**, которая подойдет под ваши нужды — например, **dst,src** или **src,src**. Списки пар IP/порт и сеть/порт позволяют вместе с портом также уточнить протокол, что может сэкономить некоторое время при работе с протоколами, которые используют сразу TCP и UDP, например, DNS или SIP. Если бы нам нужно было разрешить DNS-запросы и передачу зон с некоторых узлов, мы бы использовали список вроде такого:

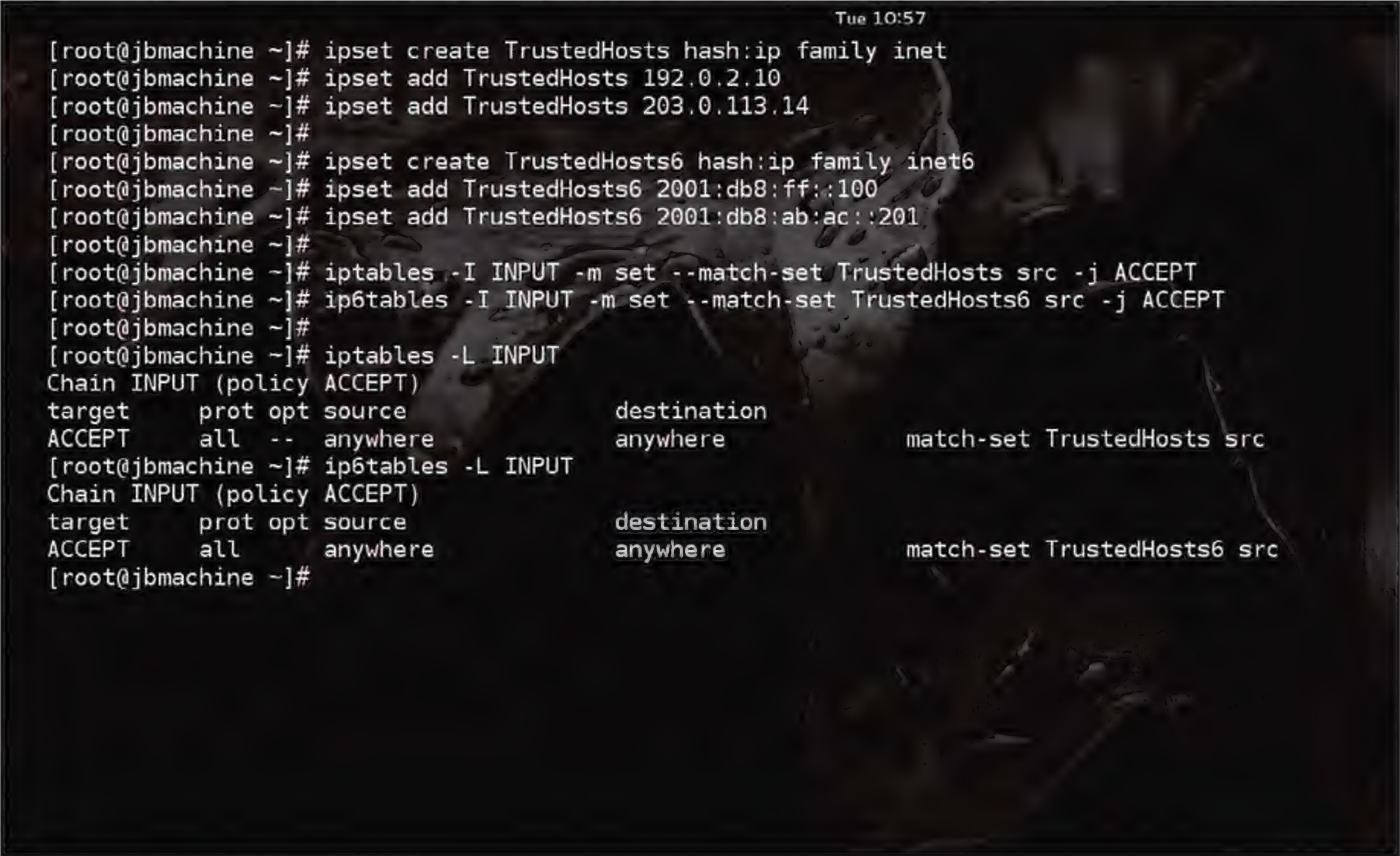
```
# ipset create DNS hash:ip,port
# ipset add DNS 192.0.2.200,udp:53
# ipset add DNS 192.0.2.200,tcp:53
# iptables -A FORWARD -m set --match-set DNS dst,dst
```

Как и *iptables*, *ipset* позволяет загружать настройки из файла и сохранять их в формат, пригодный для загрузки, следующим образом:

```
# ipset save > /путь/к/настройке/ipset.save
# ipset restore < /путь/к/настройке/ipset.save
```

Надеюсь, *ipset* поможет вам содержать настройку брандмауэра чистой, выразительной и гораздо более простой. Если вы хотите получить больше информации, можете посетить сайт проекта по адресу <http://ipset.netfilter.org> и ознакомиться с ман-страницами документации, включёнными в пакет. **LXF**

» Группы доверенных и не очень доверенных хостов как для IPv4, так и для IPv6 создавать довольно легко.



» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

СВОБОДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ВИРТУАЛИЗАЦИЯ: ЭКОНОМИЯ НА СТОИМОСТИ IT-ИНФРАСТРУКТУРЫ ДО 90%



Единая инфраструктура
на базе свободного
программного обеспечения

Нет лицензионных платежей —
расходы только на внедрение
и техническую поддержку

Минимальные затраты
на оборудование
за счет виртуализации



[www.linuxcenter.ru/shop/
linux-software/office/kitezh](http://www.linuxcenter.ru/shop/linux-software/office/kitezh)

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center

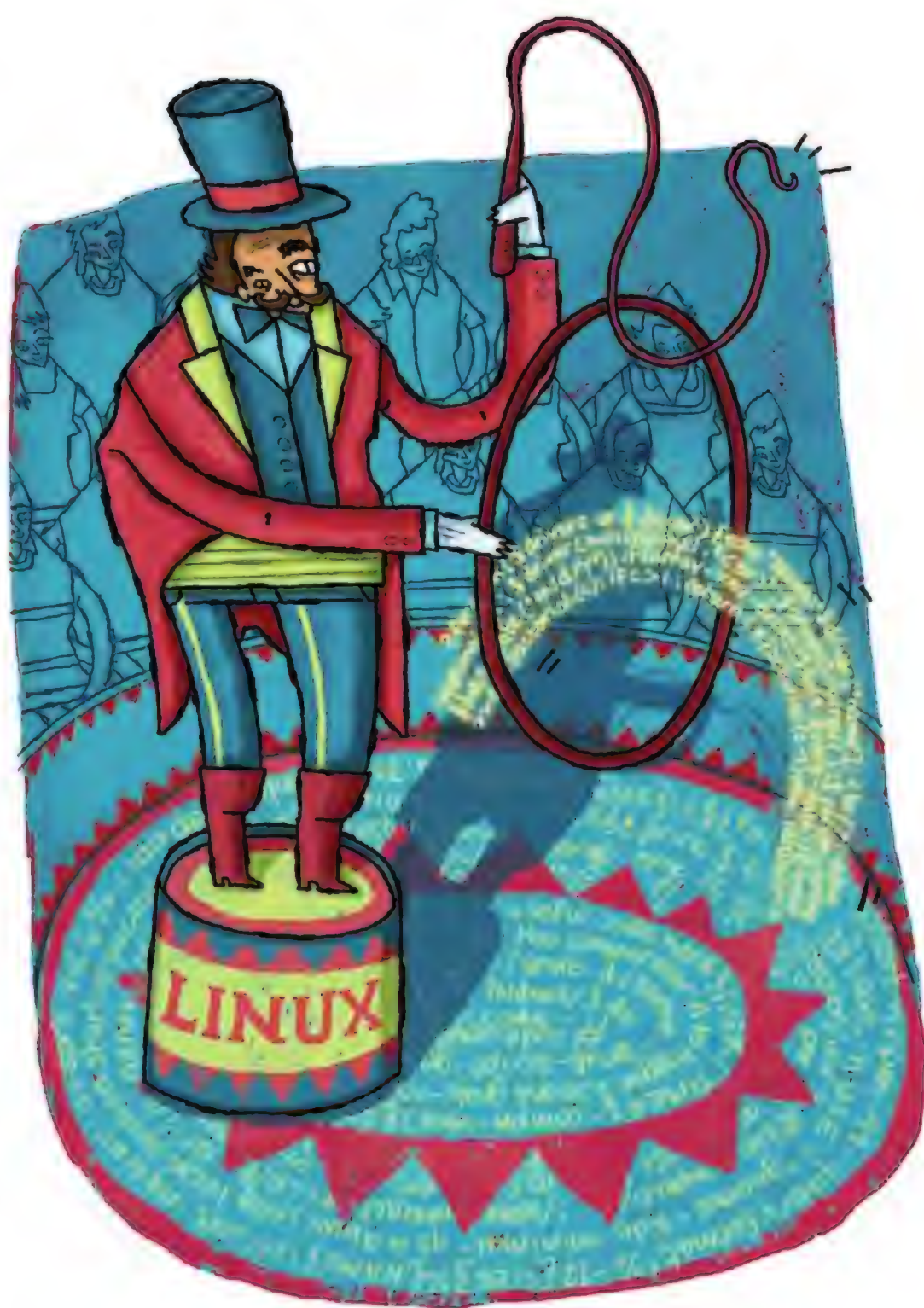
Awk: Обработка текстовых файлов для крутых

Эндрю Моллетт обрабатывает данные из файлов лога на сервере с помощью потокового редактора *sed* для простых задач и языка Awk для сложных.



Наш эксперт

Эндрю Моллетт — преподаватель Linux с более чем 700 видео на YouTube (<http://bit.ly/UrbPeng>). Также вы найдёте его курсы на www.pluralsight.com.



Если вы читали недавние выпуски *Linux Format*, то Awk должен быть вам знаком, потому что Нейл Ботвик уже давал отличное введение в этот язык [см. Учебники **LXF190**, стр. 82]. В этой статье мы рассмотрим, каким практичным он может быть для обработки журналов сервера и конфигурационных файлов.

Введение в обработку текста

Прежде чем мы начнём, позвольте продемонстрировать мощь обработки текста быстрым примером с утилитой *grep*. Вы, вероятно, уже знаете, что можете увидеть список определённых функций оболочки командой **declare -f**. Когда вы запускаете эту команду, она выводит список полных определений функций с именами. Мы можем использовать **declare -F**, чтобы вывести только имена,

но при этом немного раздражает, что вывод включает **declare -f** перед каждым именем. Команда **grep** может профильтровать вывод для нас. Просто напишем так:

```
declare -f | grep ^[a-z_]
```

Мы берём стандартный вывод команды **declare -f** и фильтруем его с помощью *grep*. Регулярное выражение, которое мы используем, определяет, что мы выводим только строки, начинающиеся (об этом говорит символ-крышка) с буквы в нижнем регистре или знака подчёркивания (об этом говорит выражение в квадратных скобках). Код внутри функции имеет отступ, а стало быть, не начинается с буквы или подчёркивания. Фильтр пройдут только строки, включающие названия, поэтому вывод будет точно таким, каким мы хотели — списком имён функций.

Применим sed для Dockerfile

Перейдём к более сложному примеру. Но мы не сразу нырнем в Awk, а используем утилиту *sed* (Stream EDitor — «потоковый редактор»), чтобы добавить возможностей файлам Dockerfile. Как объяснял Джолион Браун в прошлом выпуске [см. Учебники **LXF191/192**, стр. 74], Dockerfile используется для сборки образа Docker. Мы можем, например, взять за основу стандартный образ Ubuntu и добавить к нему сервер SSH. Или начать с образа CentOS и установить туда *Apache*.

Так или иначе, в обоих случаях надо редактировать файл настройки для данного сервиса. Давайте взглянем на Dockerfile, который может быть использован для создания контейнера SSH-сервера:

```
FROM ubuntu
RUN apt-get update && apt-get install -y openssh-server
RUN mkdir /var/run/ssh
RUN echo 'root:Password1' | chpasswd
RUN sed -i 's/PermitRootLogin without-password/PermitRootLogin yes/' /etc/ssh/sshd_config
RUN sed -i 's@session\s*required\s*pam_loginuid.so@ session optional pam_loginuid.so@g' /etc/pam.d/ssh
EXPOSE 22
CMD ["/usr/sbin/sshd", "-D"]
```

В процессе сборки нашего нового образа мы используем две строки RUN, которые выполняют код *sed*. Они обе применяют команду **sed**, чтобы заменить одну текстовую строку на другую, только с немного разным форматированием.

Скорая помощь

Чтобы узнать, какую версию *sed* вы используете, запустите команду **sed --version**. С Awk аналогично работает команда **awk --W version**.

В первом случае используется традиционный синтаксис с прямыми слэшами — ограничителями первой строки, которая будет заменена на вторую. Базовый синтаксис для замены текста таков:

```
sed -i 's/Строка/Замена/' /etc/ssh/sshd_config
```

Использование опции **-i** позволяет редактировать сам файл вместо отсылки его в **STDOUT**. Как видите, мы ищем строки в файле **/etc/ssh/sshd_config** и заменяем все те, что содержат **PermitRootLogin without-password**, на **PermitRootLogin yes**. Исходная строка стоит по умолчанию и запрещает суперпользователю **root** вход по паролю. В таком случае **root** может войти, только используя авторизацию по публичному ключу. А мы хотим сделать, чтобы **root** мог входить по паролю, который мы задали ранее в **Dockerfile**. Используя **sed** таким способом, очень легко внести необходимую правку в конфигурацию.

Во втором примере из того же **Dockerfile** мы видим, как можно использовать другие разделители, а не только прямой слэш. В данном случае мы взяли символ **@**. Мы выбрали его, так как в регулярном выражении фигурирует обратный слэш, и использование альтернативного разделителя упрощает чтение выражения. Базовый синтаксис теперь становится таким:

```
sed -i 's@Строка@Замена@' /etc/pam.d/sshd
```

Взглянув на рабочий пример из **Dockerfile**, мы увидим употребление регулярного выражения для описания строки, которая подлежит замене:

```
session\s*required\s*pam_loginuid.so
```

Здесь **\s** соответствует любому непечатаемому символу, а ***** мы используем для указания, что сюда подпадает любое число таких символов. Тем самым мы предусмотрели случаи, когда, скажем, между словами стоит два пробела или другие разделители — например, табуляция. Это регулярное выражение всё равно сработает, невзирая на тип непечатаемого символа. Строка замены читается проще: это обычная строка с обычными пробелами. Цель замены **PAM**-файла в данном случае — убедиться, что соединение всё-таки выполняется, даже когда нужен вход по паролю. Это минимальная конфигурация, поэтому все остальные элементы могут отсутствовать; отметка модуля как опционального означает, что нам не принципиальна удача или ошибка в модуле **PAM**.

Как видно, применение **sed** в этом случае даёт относительно простой механизм редактирования файлов настройки в процессе сборки образа **Docker**. Когда изменения минимальны, это предпочтительный способ создания новых конфигураций при сборке.

Аналогично можно удалять строки из файла; разве что понадобится команда **d** (от слова **delete** — «удалить») вместо **s** (от **substitute** — «заменить»). Однако при удалении нужно указать диапазон строк, с которыми мы будем работать, тогда как ранее мы имели дело со всеми строками файла. Диапазон указывается перед **d**, внутри прямых слэшей. Здесь разделителем обязан быть прямой слэш, в отличие от прошлого случая с заменой.

В следующем примере мы создадим образ **Docker**, используя базовую установку **CentOS**, установим в неё **HTTP**-сервер **Apache** и удалим ненужный модуль из конфигурации **web**-сервера:

```
FROM centos:centos6
RUN yum install -y httpd
RUN sed -i 'LoadModule\s*userdir_module/d' /etc/httpd/conf/httpd.conf
RUN echo «Добро пожаловать на Мой Сайт» > /var/www/html/index.html
EXPOSE 80
ENTRYPOINT ["/usr/sbin/httpd", "-DFOREGROUND"]
```

Конечно, не все из вас используют **Docker** — по крайней мере, прямо сейчас. (Я уверен, что со временем мы сумели бы убедить

```
#manycastserver 239.255.254.254          # manycast server
#manycastclient 239.255.254.254 autokey # manycast client

# Enable public key cryptography.
#crypto

includefile /etc/ntp/crypto/pw

# Key file containing the keys and key identifiers used when operating
# with symmetric key cryptography.
keys /etc/ntp/keys

# Specify the key identifiers which are trusted.
#trustedkey 4 8 42

# Specify the key identifier to use with the ntpdc utility.
#requestkey 8

# Specify the key identifier to use with the ntpq utility.
#controlkey 8
```

» В стандартном файле **ntp.conf** в **CentOS** много пустых строк и строк комментариев, так что за деревьями не видно леса. Ситуация исправима — с помощью **sed**.

вас в его достоинствах.) Впрочем, для **sed** есть множество других применений.

Один из способов, которым я часто применяю **sed** — это с добавкой **-i**, для редактирования на месте, с расширением, создающим резервную копию перед изменением. Многие файлы настройки в **Linux** наполнены комментариями и пустыми строками. Хотя я не против комментариев, они могут усложнять понимание конфигурации — и, в некоторых случаях, приводить к дублированию настроек: ведь не так просто найти, где настройка была задана ранее.

Простая иллюстрация — файл **/etc/ntp.conf**. Это конфигурация сервера времени, и на моём сервере с **CentOS 6** она содержит 53 строки; а при деле только 11 из них, остальное — комментарии. Это не самый крайний случай, но он демонстрирует проблему. Я бы создал резервную копию этого файла, чтобы на всякий случай иметь прокомментированный пример; ну, а «очищенный» оригинал станет рабочей конфигурацией:

```
sed -i commented '/^#/d;/^$/d' /etc/ntp.conf
```

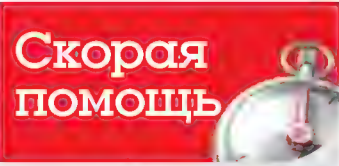
Здесь **sed** использует два выражения, разделённые точкой с запятой (;). Первое выражение удалит строки, начинающиеся с **#** — то есть закомментированные. Второе выражение удалит пустые строки, то есть (в соответствии с регулярным выражением **^\$**) те, которые начинаются символом конца. Запустив эту команду от имени **root**, мы урежем содержимое **ntp.conf** до 11 строк и сохраним исходную версию файла. Она, с нетронутыми комментариями и лишними строками, теперь называется **/etc/ntp.conf.commented**.

Обратите внимание на расширение: оно следует сразу за опцией **-i**. Между опцией и желаемым расширением не должно быть лишних пробелов.

Невероятный Awk

Если **sed** — это старший брат **grep**, то **Awk** можно назвать отцом их обоих. В прошлой статье [Учебники **LXF190**, стр. 82] Нейл познакомил вас с **Awk** и его возможностями. Сегодня мы применим эти возможности на практике. Первым делом посмотрим, как с помощью **Awk** можно улучшить вывод команды **lastlog**, потом перейдём к работе с **XML**, а затем к обработке больших текстовых файлов с целью сделать выжимку из журналов сервера.

Чтобы начать, требуется хорошее знакомство с **lastlog**. При вызове **lastlog** без аргументов она выводит время последнего входа со всех учетных записей, включая служебные, с которых вообще никто никогда не входит. Такой вывод, мягко выражаясь,



Когда вы используете **sed -i** для редактирования файла, бывает разумно сначала опустить параметр **-i** и убедиться, что вывод соответствует желаемому.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe!

грязноват. Либо, мы можем запускать команду с опцией, чтобы показать время последнего входа только для одного пользователя: **lastlog -u bob**, например. Ещё один вариант — показать только учетные записи, которые не входили в систему за последние 90 дней: **lastlog -b 90**.

Это замечательно, но учетные записи, сроду не входившие в систему, всё ещё показаны. В идеале хотелось бы иметь отчёт только из имени и даты входа, исключаящий учетные записи, отмеченные как никогда не входившие.

Сначала мы просто попросим Awk отфильтровать не входившие учетные записи. Это не очень отличается от действий с *grep*, зато проиллюстрирует, как Awk применяется для «обратного поиска»:

```
lastlog | awk '!/Never/ { print }'
```

Мы посылаем вывод **lastlog** прямо в Awk. Выражение Awk начинается с шаблона. Мы делаем шаблон «обратным» по отношению к строкам, содержащим подстроку *Never*; иными словами, исключаем строки, содержащие *Never*. Затем тело Awk просто печатает каждую строку, подошедшую под шаблон, так что мы увидим все учетные записи, с которых заходили в систему хотя бы раз.

Можно расширить этот шаблон, исключив также пользователей *root* и скрыв заголовок **Username**:

```
lastlog | awk '!(/Never/ || /^root/ || /^Username/) { print }'
```

Скобки использованы, чтобы сгруппировать три шаблона вместе и применить отрицание к ним единожды. Две вертикальные полосы (*||*) обозначают логическое ИЛИ. Мы не обрабатываем строки, которые либо содержат *Never*, либо начинаются с *root* или с *Username*. Хотя эти множественные исключения и могли быть записаны как выражение *grep*, мы уже достигли территории, где с Awk достичь нужного результата проще.

То, что мы сделали, неплохо для одной строчки кода в терминале, но если желателен реально полезный результат, надо действовать с Awk смелее.

Для начала создадим файл Awk, что уменьшит объем синтаксиса при вводе в терминал ради повторного использования команды. Это продемонстрирует несколько ценных техник Awk. Файл, с которым мы будем работать, выглядит так:

```
BEGIN {
    printf «%8s %11s\n»,»Пользователь»,»Время входа»
    print «=====»
}
!(/Never logged in/ || /^Username/ || /^root/) {
    cnt++
    if (NF == 8)
        printf «%8s %2s %3s %4s\n», $1,$5,$4,$8
    else
        printf «%8s %2s %3s %4s\n», $1,$6,$5,$9
    }
END {
    print «=====»
    print «Обработано пользователей: », cnt
}
```

» Удалять излишки с *sed* легко; но легко и переборщить — поэтому сначала тестируйте без опции *-i*, и файл останется нетронутым. Вывод попадет только в консоль.

```
[ root@centos6 ~]$
[ root@centos6 ~]$
[ root@centos6 ~]$ sed -i.commented '/^#/d;/^$/d' /etc/ntp.conf
[ root@centos6 ~]$ ls /etc/ntp.conf*
/etc/ntp.conf /etc/ntp.conf.commented
```

Чтобы запускать его, нужно находиться в той же папке, что и файл. Вот два примера его применения:

```
lastlog | awk -f lastlog.awk
```

```
lastlog -b 60 | awk -f lastlog.awk
```

Первый пример обрабатывает всех пользователей; второй — только тех, которые не входили в систему за последние 60 дней. Вы уже можете оценить могущество Awk и его обработки данных, когда сравните вывод **lastlog -b 60** с этим примером.

Сам файл Awk состоит из трёх разделов. Первый и последний называются вполне уместно: **BEGIN** и **END**. Основной, безымянный раздел — это тело программы. Разделы **BEGIN** и **END** выполняются лишь один раз, тогда как основное тело выполняется для каждой строки, соответствующей шаблону.

В разделе **BEGIN** мы можем при необходимости установить переменные — например, разделители; или, как в данном случае, информацию о заголовке. Использование **printf** вместо **print** облегчает форматирование информации для наших нужд.

Завершающий раздел **END** обычно содержит вывод итоговой информации. Здесь мы выводим число пользователей, взяв его из переменной **cnt**, которая увеличивалась в теле.

Наконец, к телу. Здесь видно много элементов, предоставляемых языком Awk. Само тело определено внутри фигурных скобок. Непосредственно перед ними задаётся шаблон, о котором мы говорили ранее. Тело работает только с теми строками, которые подпали под критерии в этом шаблоне.

Первые строки тела нашей программы определяют и увеличивают переменную **cnt**. Мы используем её как счётчик, чтобы вывести в **END**. Во время первой итерации она не определена и поэтому имеет значение 0, которое мы увеличиваем, устанавливая в 1. Следующее совпадение с шаблоном сделает её равной 2, и так далее.

Мы написали строку **If (NF == 8)**, чтобы удостовериться, что напечатаем нужное поле. Логины с удалённых клиентов имеют девять полей, а с локальных консолей — только восемь. Число полей в строке хранится в переменной **NF**. Наши выражения печатают соответствующие поля в зависимости от того, восемь или девять полей мы получили.

Awk для обработки данных XML

Теперь посмотрим, как Awk может обрабатывать данные в XML. Попутно мы узнаем, что хотя по умолчанию записи, с которыми работает Awk — это строки, мы можем установить переменную **RS**, чтобы запись содержала больше одной строки.

В данном сценарии мы храним информацию о виртуальных хостах web-сервера Apache в едином файле настройки, но хотим иметь возможность вывести полные записи о любом заданном виртуальном хосте. Виртуальные хосты начинаются с открывающего тэга наподобие `<VirtualHost *:80>` и закрываются тэгом `</VirtualHost>`. Чтобы пример заработал, нужно обеспечить пустые строки между концом прошлого и началом следующего хоста. Если это не так, можно при помощи *sed* вставить пустую строку после каждого `</VirtualHost>`. Предположим, что все хосты определены в файле **virtualhost.conf**, но пустые строки присутствуют

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.


```
andrew@pxepi ~ $ lastlog | awk '!/Never/ { print }'
```

Username	Port	From	Latest
root	tty1		Wed Jun 19 12:38:22 +0000 2013
pi	pts/0	192.168.0.57	Tue Nov 26 11:16:46 +0000 2013
andrew	pts/0	192.168.0.58	Sat Nov 1 14:05:06 +0000 2014

» С Awk легко эмулировать другие инструменты, такие как *grep*.

не всегда. Следующий код отредактирует файл, добавив в него пустые строки:

```
sed -i '/<\VirtualHost>/G' virtualhost.conf
```

Пример файла с виртуальными хостами, с которым мы работаем, выглядит так:

```
<VirtualHost *:80>
DocumentRoot /www/example
ServerName www.example.org
# Другие директивы
</VirtualHost>

<VirtualHost *:80>
DocumentRoot /www/theurbanpenguin
ServerName www.theurbanpenguin.com
# Другие директивы
</VirtualHost>

<VirtualHost *:80>
DocumentRoot /www/linuxformat
ServerName www.linuxformat.com
# Другие директивы
</VirtualHost>
```

Теперь у нас есть файл в правильном формате, и для поиска нужных записей мы можем использовать такой файл Awk, **vh.awk**:

```
BEGIN { FS = "<\VirtualHost>"; RS="\n\n"; }
$0 ~ searchstring { print }
```

Блок **BEGIN** устанавливает разделителем полей закрывающий тэг VirtualHost. По нему будут определяться границы записей. Запись — обычно строка, но мы сделали так, чтобы она ограничивалась двумя последовательными переводами строки. Основное тело печатает записи, которые теперь включают полное определение виртуального хоста, сравнивая каждую запись (**\$0**) с переменной, которую мы будем задавать при запуске (**searchstring**). Команда для запуска Awk будет примерно такой:

```
awk -f vh.awk searchstring=www.example.org virtualhost.conf
```

Заметьте, что мы передаём переменной значение во время выполнения команды. Соответствующим результатом будет

```
<VirtualHost *:80>
DocumentRoot /www/example
ServerName www.example.org
# Другие директивы
</VirtualHost>
```

Анализируем логи с Awk

Напоследок рассмотрим, как задействовать мощь Awk для чтения серверных логов и вывода количества заходов каждого клиента. Первое поле в логе содержит IP клиента. Можно задействовать массивы в Awk, чтобы посчитать обращения от каждого клиента. Мы будем работать с файлом из более 30 000 строк — пример из реальной жизни.

Нам снова понадобится файл Awk — как мы могли убедиться, это вполне нормально. На сей раз назовём его **count.awk**:

```
BEGIN {
print «Журнал доступа»
}
{ ip[$1]++ }
END {
```

```
for (i in ip)
print i, " обращался ", ip[i], " раз." }
```

Блок **BEGIN** просто печатает заголовок. Основной блок создаёт новый массив для всех возможных значений поля 1, IP-адреса клиента. Таким образом, на каждый IP-адрес, с которого было обращение к серверу, у нас имеется именованный элемент массива (**ip**). Значение конкретного элемента увеличивается каждый раз, когда поле вновь имеет такой же вид. Основная работа на сей раз заключена в блоке **END**, который использует цикл **for** для обхода всех именованных элементов массива **ip** и вывода их значений. Команду

```
awk -f count.awk access.log
```

вы можете ожидать вывод, похожий на приведённый ниже. Поскольку данные были с настоящего сервера, мы изменили первые октеты IP-адресов клиентов:

```
xxx.157.100.28 обращался 1 раз.
xxx.180.86.233 обращался 10 раз.
xxx.241.226.216 обращался 5 раз.
xxx.99.52.100 обращался 12 раз.
```

Файл Awk легко отредактировать, чтобы отобразить коды доступа HTTP, это поле 9 из лога сервера. Тогда мы увидим количество обращений к серверу за весь период, который отражён в логе. С моим логом вывод получился таким:

```
Журнал доступа
Статус 200 случался 23825 раз.
Статус 206 случался 48 раз.
Статус 301 случался 60 раз.
Статус 302 случался 21 раз.
Статус 304 случался 2279 раз.
Статус 403 случался 131 раз.
Статус 404 случался 4386 раз.
Статус 501 случался 66 раз.
```

Ошибки 403 — это запрещённая активность, для которой должна была сработать, но не сработала безопасность; ошибки 404 — это, как всем, наверное, известно, «Страница не найдена»; коды 2xx — успешные обращения; 3xx — обычно перенаправления; а 5xx связаны с ошибками CGI. Обработка 30 000 строк заняла у Awk секунды, что показывает, как легко мы можем начать анализировать информацию. **LXF**

```
File Edit View Terminal Tabs Help
dal:~/Desktop # lastlog -b 60 | awk -f lastlog.awk
Username Login date
=====
    geeko  2 Jul 2013
    bob    6 Jul 2013
=====
Total Number of Users Processed:  2
dal:~/Desktop #
```

» Поняв, как полезен Awk для настройки вывода команд под ваши нужды, вы создадите с его помощью набор собственных инструментов.

Ваш домашний процессор

Михаил Остапкевич решил сделать ёлочную гирлянду-мигалку, а Евгений Балдин вспоминает Новый год.



Наш эксперт

Михаил Остапкевич
Романтик, очарованный компьютерами и создаваемыми в них идеальными мирами.



Наш эксперт

Евгений Балдин
Физик, который действительно знает, что такое нехватка вычислительных ресурсов.



В LXF189 мы пристыковывали ПЛИС к компьютеру через USB. Теперь с помощью ПЛИС решим более интересную задачу: построим программную реализацию процессора. Конечно, это будет не процессор уровня x86_64 или ARM64. Это всё-таки слишком сложно, хоть и тоже интересно. Сейчас наша цель — построить максимально простой процессор и в честь Нового года, прошедшего или будущего (надо же начинать готовиться заранее), написать для него программу управления ёлочной гирляндой из RGB-светодиодов. Чтобы получить красивые световые эффекты, процессору необходимо уметь хранить таблицы чисел и производить разнообразные вычисления над ними.

А что было до нас?

К настоящему моменту разработано большое число самых разнообразных программных реализаций процессоров на ПЛИС. Как правило, вместо построения собственного велосипеда лучше воспользоваться именно ими. Далее перечислены некоторые из них.

» **Angelus Research Forth Processor** Одна из наиболее лаконичных реализаций многочисленного семейства форт-процессоров; разработана около 15 лет назад. Сам язык Форт [Forth] был создан в конце 1960-х для автоматизации управления радиотелескопом,

то есть он заведомо достаточно мощен, чтобы управлять ёлочной гирляндой. Доступны исходные тексты на VHDL (<http://www.ultratechnology.com/4thvhd.html>). Код не является свободным, но для своих разработок его можно использовать. Объём исходного кода составляет всего 14 килобайт. Разрядность шины данных — 16 бит, как и разрядность адресной шины. Эта реализация форт-процессора имеет два аппаратно реализованных стека, а именно, стек вызовов и стек данных. Глубина стека возвратов равна 16. Так как доступен исходный текст, то все эти параметры можно в определённых пределах варьировать. Для стеков обрабатываются ошибки их переполнения или недополнения. Наряду с арифметическими командами (plus, minus, times, divide), этот процессор реализует команды работы со стеком в стиле стандартных слов Форта (drop, dup, rot, swap). Кроме них, есть управляющие команды, команды сравнения (equal, zero_equal, greater_than) и доступа к памяти (store). Всего в процессоре реализовано примерно 24 команды. Другой пример форт процессора с VHDL-описанием на русском языке доступен по адресу <http://habrahabr.ru/post/149686/>. В более широком контексте уместно упомянуть такие ASIC-реализации форт-процессоров, как ультраэкономный 4-битный Atmel MARC4 (<http://www.atmel.com/Images/doc4747.pdf>) или многоядерный и тоже экономный Green Arrays GA144 (<http://www.greenarraychips.com/>).

» **Процессор ZPU** Альтернатива семейству форт-процессоров; распространяется под лицензиями GPL и BSD. ZPU (<http://opensource.zylin.com/zpu.htm>) — это самый маленький 32-битный процессор, который поддерживается GCC. Программная архитектура ZPU воплощена в нескольких реализациях и имеет 0-адресную (стековую) архитектуру. Отнести процессор однозначно к RISC или CISC нельзя. Базовая часть команд должна быть реализована в аппаратной логике (как у RISC), а остальные могут тоже быть аппаратными или реализованы в микрокоде. Конкретные детали варьируются для разных реализаций ZPU. Более миниатюрные реализации в большей степени используют микрокод, а в более производительных превалирует аппаратная реализация. То есть в зависимости от задачи и аппаратуры можно выбирать как компактные, хотя и медленные реализации, так и более объёмные и соответственно более производительные.

Как и в Angelus, ZPU основан на интенсивном использовании стека. Арифметические команды, реализующие двоичные операции (ADD, DIV, MOD, MULT, SUB) снимают два верхних элемента со стека, выполняют операцию и помещают результат обратно в стек. Аналогично работают и команды битовой арифметики (AND, FLIP, NOT, OR, XOR). Команды сравнения (LESSTHAN, LESSTHANOREQUAL, ULESSTHAN, ULESSTHANOREQUAL) возвращают на стек логическое значение — результат сравнения (0 — ложь, 1 — истина).

» **OpenRISC 1000 (OR1K)** Это более основательная разработка для высокопроизводительных встраиваемых систем (http://opencores.org/or1k/Architecture_Specification). Все исходные тексты доступны под лицензией LGPL. Естественно, поддерживается инструментарий *GCC*. OpenRISC позволяет запускать на себе GNU/Linux, причём разрядность архитектуры может быть 32 или 64 байта, и, в полном соответствии с названием, это архитектура RISC. Также OpenRISC имеет выделенные инструкции для работы с памятью, поддерживает плавающую запятую, векторные расширения, виртуальную память с механизмом защиты страниц, аппаратную поддержку многопоточности, быстрое переключение контекста, конвейер с механизмом предсказания переходов, поддержку SMP и многое другое.

Конечно, существует огромное число и других реализаций процессоров (b16, KCPSM, MicroCore, MicroPlaze, openRISC 1200, PicoBlaze), но и описанных выше вполне достаточно для того, чтобы обуздать ёлочную гирлянду.

Требования к процессору

Первое, что нужно от нашего процессора — это получить простую и понятную реализацию. Поэтому назовем его Simple Processor Unit, или кратко SPU2014. Ведь мы её потом будем использовать как основу для других разработок. Это, в частности, означает, что мы не будем требовать от SPU2014 широкого набора функций, эффективности реализации и высокой производительности.

Кроме простоты реализации, нам нужна и простота использования. В минимальном варианте процессор можно использовать без всего привычного для современного программиста инструментария (компилятора, ассемблера и т.д.), имея лишь терминальную программу для доступа к плате Papilio. Для этого процессор должен иметь возможность выполнения не только целых программ, но и одиночных команд, поступающих через последовательный порт. То есть нужен интерактивный режим, как это было в старых программируемых калькуляторах МК-61/МК-52.

Перейдём к функциональным требованиям. Точнее, для данной проблемной области это можно назвать архитектурой или программной архитектурой процессора. Архитектура определяется в первую очередь форматами данных, набором регистров, набором инструкций, режимами функционирования, организацией памяти, прерываниями. Наши форматы данных — это целое число и адрес. Оба имеют размер 16 бит. Для этого потребуются следующие регистры:

- 1 Аккумулятор для хранения аргумента унарных операций и одного из аргументов бинарных операций.
- 2 Вспомогательный регистр для хранения второго аргумента.
- 3 Регистр результата.
- 4 Регистр счётчика команд.
- 5 Регистр вершины стека.

Теперь опишем минимальный набор нужных инструкций. В первых, нам необходимы команды чтения и записи оперативной памяти:

- » **read** Чтение из памяти в регистр аккумулятора.
- » **write** Запись из регистра аккумулятора в память.
- » **push** Проталкивание из аккумулятора на вершину стека.
- » **pop** Выталкивание с вершины стека в аккумулятор.

Все остальные команды будут работать только с регистрами. Вторая группа команд:

- » арифметические (+, -),
- » сравнения (=, <, >, #).

Следующая группа — это команды перехода:

- » **jmp** команда безусловного перехода (go to),

- » **call** вызов подпрограммы,

- » **ret** возврат из подпрограммы (return),

- » **jz** команда условного перехода,

- » **Битовые команды** По аналогии с битовыми операциями C (!, &, |, ^).

Для ввода числовых значений используем команды,двигающие один шестнадцатеричный разряд:

- » (0..9, a..f).

Ещё одна группа — команды пересылки между регистрами:

- » ; Копирование из регистра аккумулятора во вспомогательный регистр.

- » @ Копирование из регистра результата в аккумулятор.

Для управления внешними устройствами нужны команды ввода/вывода:

- » **in** Ввод данных из порта (input).

- » **out** Вывод данных в порт (output).

- » **wait** Задержка.

Последняя группа — вспомогательные команды, используемые в интерактивном режиме:

- » **run** Используется для запуска программы с адреса, указанного в аккумуляторе.

- » **halt** Для остановки выполнения программы.

- » **команды [и]** Служат для входа в режим записи программы и выхода из него.

Будут доступны следующие режимы функционирования:

- » интерактивный режим;
- » режим ввода программы;
- » режим выполнения программы.

В интерактивном режиме пользователь с помощью терминальной программы посылает процессору единичные команды. Из этого режима можно перейти в любой из двух других режимов. В режиме ввода программы пользователь может записать фрагмент программы (всю программу или некоторую процедуру) в непрерывный блок оперативной памяти, начинающийся с указанного в аккумуляторе адреса. По окончании ввода происходит переход в интерактивный режим.

В режиме выполнения программы процессор выполняет поток команд, считываемых из оперативной памяти до выполнения команды останова или прерывания программы пользователем. После этого происходит возврат в интерактивный режим. Прерывания и исключения в этой реализации процессора не предусмотрены.

Реализация

Заимствуем из предыдущей статьи все модули, кроме управляющего. Новая реализация управляющего модуля будет центральной в реализации процессора. Исходные коды можно найти по адресу <http://qwertus.com/fpga/p2.html> или на диске, идущем с журналом. Модули передачи данных через последовательный порт слегка доработаем и введём буферизацию передаваемых данных. Совсем немного изменится и модуль верхнего уровня, описывающий сборку всех остальных модулей. Также добавим модуль реализации оперативной памяти.

» Управляющее устройство

Весь исходный текст приведен на сайте <http://qwertus.com/fpga/p3.html>. Рассмотрим здесь только некоторые характерные его фрагменты. Регистровый файл представляет собой набор 8-битных векторов:

```
-- аккумулятор
signal reg_acc : std_logic_vector (7 downto 0) := «00000000»;
-- вспомогательный, для второго операнда
```

»

» **Не хотите пропустить номер?** Подпишитесь на www.linuxformat.ru/subscribe!


```

signal reg_aux : std_logic_vector (7 downto 0) := «00000000»;
signal reg_res : std_logic_vector (7 downto 0) := “00000000”;
-- счётчик команд
signal reg_ip : std_logic_vector (7 downto 0) := “00000000”;

```

Весь цикл обработки инструкций расположен в одном блоке process. На верхнем уровне этого блока реализован автомат с фазами обработки команды. Главные фазы:

- » простой;
- » запуск выполнения;
- » завершение выполнения.

Также есть несколько состояний для команд чтения и записи памяти. Выход из состояния простоя происходит либо при получении новой интерактивной команды от пользователя (когда iexec_req переходит в единичное состояние), либо в режиме исполнения программы (когда mode_run в единичном состоянии).

В первом случае, когда код команды получен, то сразу происходит переход в состояние запуска исполнения. Во втором случае необходимо считать код по адресу, указанному в счетчике команд, и инкрементировать этот счётчик. Следом за описанием фазы простоя расположено описание фазы запуска исполнения. В ней для всех реализованных инструкций описана их логика. Вот соответствующий фрагмент кода от начала блока process до оператора множественного выбора по коду инструкции на фазе запуска исполнения:

```

process(clk_fastest)
begin
  if rising_edge(clk_fastest) then
    case iexec_state is
      when idle =>
        iexec_done <= '0';
        if iexec_req = '1' and iexec_req_handled = '0' then
          -- получена команда от пользователя
          reg_code <= iexec_code;
          -- подтвердить прием команды
          iexec_ack <= '1';
          iexec_auto <= '0';
          iexec_state <= start;
          -- исключить повторную обработку этой команды
          iexec_req_handled <= '1';
        elsif mode_run = '1' then
          -- активен режим исполнения программы
          -- сдвинуть счетчик команд на следующую команду
          reg_ip <= reg_ip + “00000001”;
          iexec_fetch <= '1';
          iexec_auto <= '1';
          -- перейти в режим начала чтения кода команды
          -- из памяти
          iexec_state <= mem_waitready_fetch;
        elsif iexec_req = '0' and iexec_req_handled = '1' then
          -- сбросить флаги после завершения интерактивного
          -- исполнения команды
          iexec_req_handled <= '0';
          iexec_ack <= '0';
        end if;
      when start => -- состояние начала исполнения команды
        case reg_code is

```

Приводить все реализации команд нет смысла: среди них есть много похожих. Рассмотрим несколько характерных. Вот пример реализации арифметической команды (вычитание):

```

when icode_sub =>
  reg_acc <= reg_aux - reg_acc;
  iexec_state <= finish;

```

» Подсистема памяти

Оперативная память может использовать либо блоковую память внутри чипа FPGA, либо внешнюю динамическую память. У Papilio One внешней памяти нет, так что остается только блоковая память. Обычно для работы с ней используется IP Core Generator. Детальное описание можно найти, например, в гл. 15 книги Майка Филда [Mike Field] *FPGA course. Introducing the Spartan 3E FPGA and VHDL* (http://hamsterworks.co.nz/mediawiki/index.php/FPGA_course). В представленном здесь коде **rppu_mem_ram1.vhd** упор сделан на понятность и прозрачность (в ущерб максимально эффективному использованию имеющегося чипа).

На входе модуля имеются

- » сигнал запроса req (переход в высокий уровень означает поступление запроса на доступ к памяти);
- » сигнал операции r_w (высокий уровень — запись, низкий — чтение);
- » шина адреса addr;
- » шина входных (записываемых) данных data_in.

На выходе модуля —

- » шина выходных (прочитанных) данных data_out;
- » сигнал готовности ready.

Приведём полностью текст модуля:

```

entity rppu_mem_ram1 is
  port(clk : in STD_LOGIC;
        req : in STD_LOGIC; r_w : in STD_LOGIC;
        addr : in STD_LOGIC_VECTOR (7 downto 0);
        data_in : in STD_LOGIC_VECTOR (7 downto 0);
        data_out : out STD_LOGIC_VECTOR (7 downto 0);
        ready : out STD_LOGIC);
end rppu_mem_ram1;
architecture Behavioral of rppu_mem_ram1 is
  signal reg_ready : std_logic := '1';
  signal req_handled : std_logic := '0';
  signal reg_data_out : std_logic_vector (7 downto 0) :=
    “00000000”;
  type rppu_memblock is array (0 to 255) of std_logic_vector (7
    downto 0);
  signal memblock : rppu_memblock;
begin
  process(clk) begin
    if rising_edge(clk) then
      if req = '1' and req_handled = '0' then
        -- поступил необработанный запрос
        -- исключаем повторную обработку запроса
        req_handled <= '1';
        reg_ready <= '0';
        if r_w = '1' then
          -- запрос на запись
          memblock(conv_integer(unsigned(addr))) <= data_in;
        else
          -- запрос на чтение
          reg_data_out <= memblock(conv_integer(unsigned(addr)));
        end if;
      end if;
      if req = '0' and req_handled = '1' then
        -- забываем факт обработки, когда запрос обработан
        req_handled <= '0';
        -- запрос обработан
        reg_ready <= '1';
      end if;
    end if;
  end process;

```

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.


```
ready <= reg_ready;  
data_out <= reg_data_out;  
end Behavioral;
```

» Интерфейс ввода/вывода

Целью интерфейса ввода/вывода является обеспечение связи с внешним миром. Во-первых, это обмен данными с хост-машиной, и для этого модули из предыдущей статьи (LXF189) могут быть использованы без изменений. Во-вторых, это связь с управляемыми процессором устройствами, то есть с несколькими гирляндами, состоящими из RGB-светодиодов, подключенных к выводам платы через ключ на базе транзистора.

Любой обмен данных процессора и внешних устройств производится через специализированные ячейки памяти, называемые портами ввода/вывода. Существует два варианта их введения в архитектуру. В одном варианте они размещены в общем адресном пространстве памяти, и специальных команд для них не требуется. Другой вариант — порты размещены в отдельном адресном пространстве; для доступа к портам тогда требуются отдельные команды. Здесь реализован второй вариант. Для работы необходимы следующие порты:

- 1 Управляющий регистр интерфейса передачи данных.
- 2 Статусный регистр интерфейса передачи данных.
- 3 Регистр данных для передачи.
- 4 Регистр данных для приёма.
- 5 Регистр состояния цифровых выводов.
- 6 Регистры состояния выводов с ШИМ.

Примеры использования

Сначала воспользуемся нашим процессором как калькулятором. Сложим шестнадцатеричные числа ab и 12 и напечатаем результат:

```
ab ; 12 + ``  
Запишем результат из аккумулятора в ячейку памяти с адресом 95:  
; 95 write
```

```
А теперь прочитаем обратно в аккумулятор:  
95 read @  
Запишем это значение в регистр состояния вывода 17 (шестнадцатеричное 11) с ШИМ:  
; 11 out
```

```
Теперь напишем первую простейшую программу, мигающую одним светодиодом на ШИМ-выводе 17 (половину времени — высокая яркость, AA; половину — низкая, но видимая, 22). Располагаем её, начиная с ячейки 00:  
00[aa; 11 out 17; 2 wait 22; 11 out 00 jmp]  
И запускаем её также с адреса 00:  
00 run
```

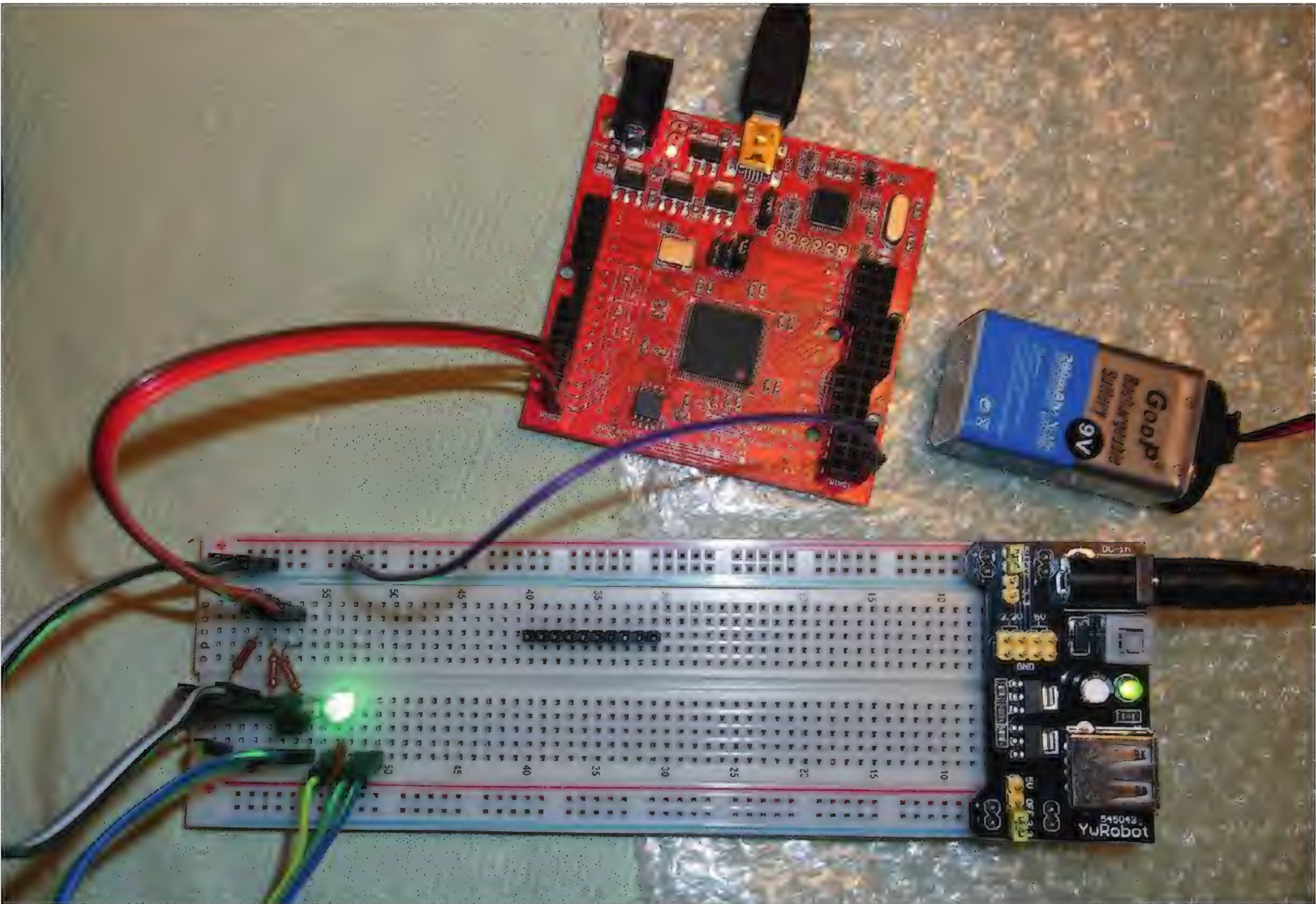
Пишем программу для SPU2014

Гирлянда будет работать в нескольких режимах. Каждому режиму соответствует реализующая его подпрограмма. Главная подпрограмма поочередно вызывает одну из них.

Полный текст всех подпрограмм приведен на сайте <http://qwertus.com/spu2014/ny1.html>. Для примера приведём одну из них.

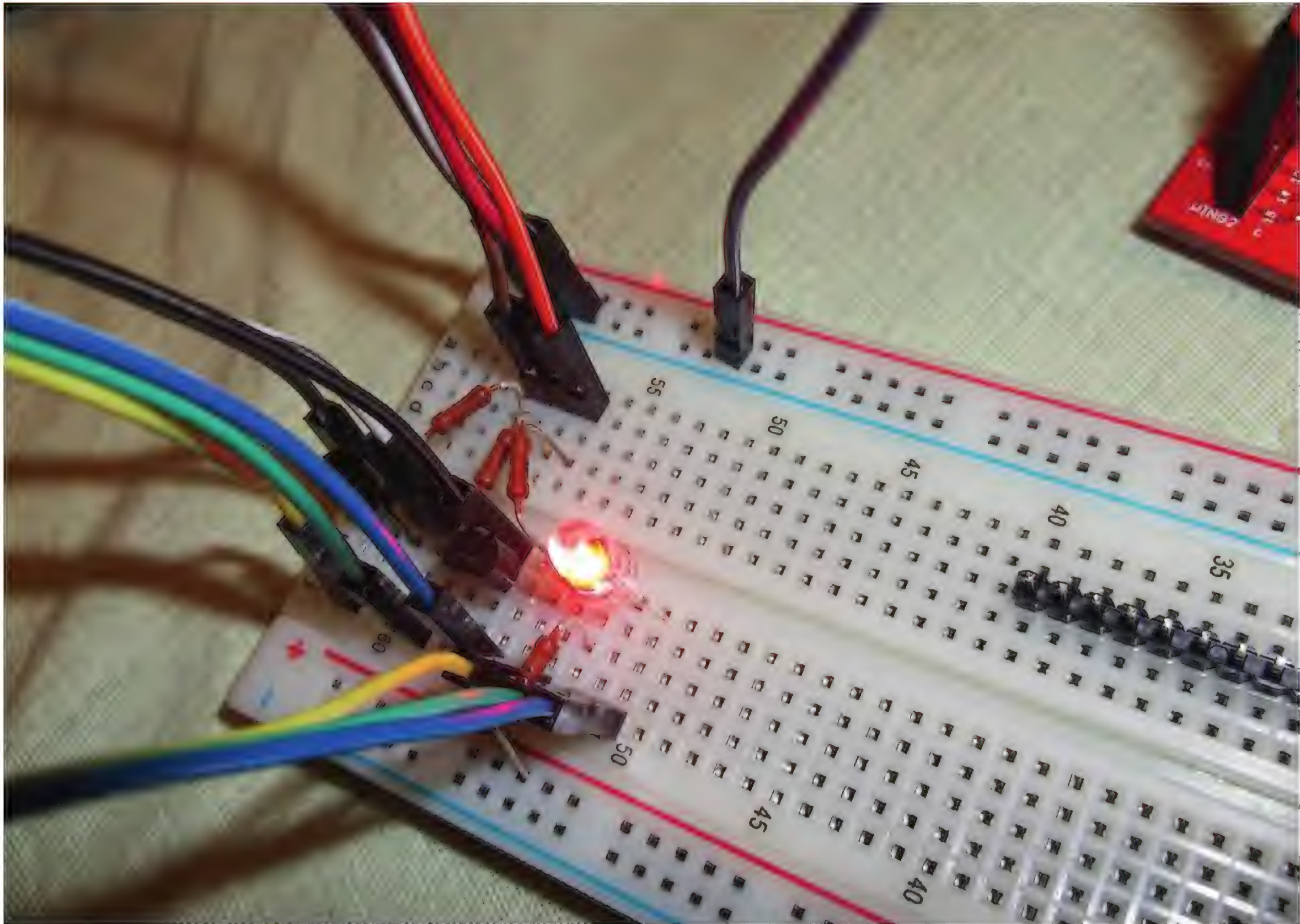
```
00[  
00; a0 write  
55; a1 write  
ab; a2 write  
30 jmp  
]  
30[  
0f; 80 out 02; 81 out 01; 82 out 00; 82 out wait  
a0 read @ ; 0f out 01 + ; a0 write  
a1 read @ ; 10 out 01 + ; a1 write  
a2 read @ ; 11 out 01 + ; a2 write  
30 jmp  
]  
Копируем программу в окно терминальной программы и запускаем её с адреса 00:  
00 run
```

»



» Рис. 1. Наш процессор полностью собран.

Рис. 2. Схема транзисторного ключа.



Конечно, читаемость чуть более сложной программы уже под большим вопросом. Всё-таки это непосредственно исполняемый код, а не исходник. Далее нам потребуется простейшая утилита ассемблера с возможностью именования меток, ячеек памяти и т.д.

Собираем гирлянду

Один RGB-светодиод можно подключать непосредственно к трём сигнальным выводам и земле на плате через ограничительные сопротивления. Если же мы хотим собрать полноценную гирлянду, то подключать их напрямую к выводам нельзя, так как их нагрузочная способность лежит в диапазоне 2–16 мА, в зависимости от того, как вывод сконфигурирован. Поэтому необходимо собрать три транзисторных ключа для каждой гирлянды. То есть, например, для трёх гирлянд получается 9 ключей. Конечно, питание в этом случае нужно внешнее. В конкретном случае для этой цели была использована недорогая макетная плата YwRobot Power MB V2. На этой плате обе перемычки, задающие выходной вольтаж, были установлены в положение 3,3 В.

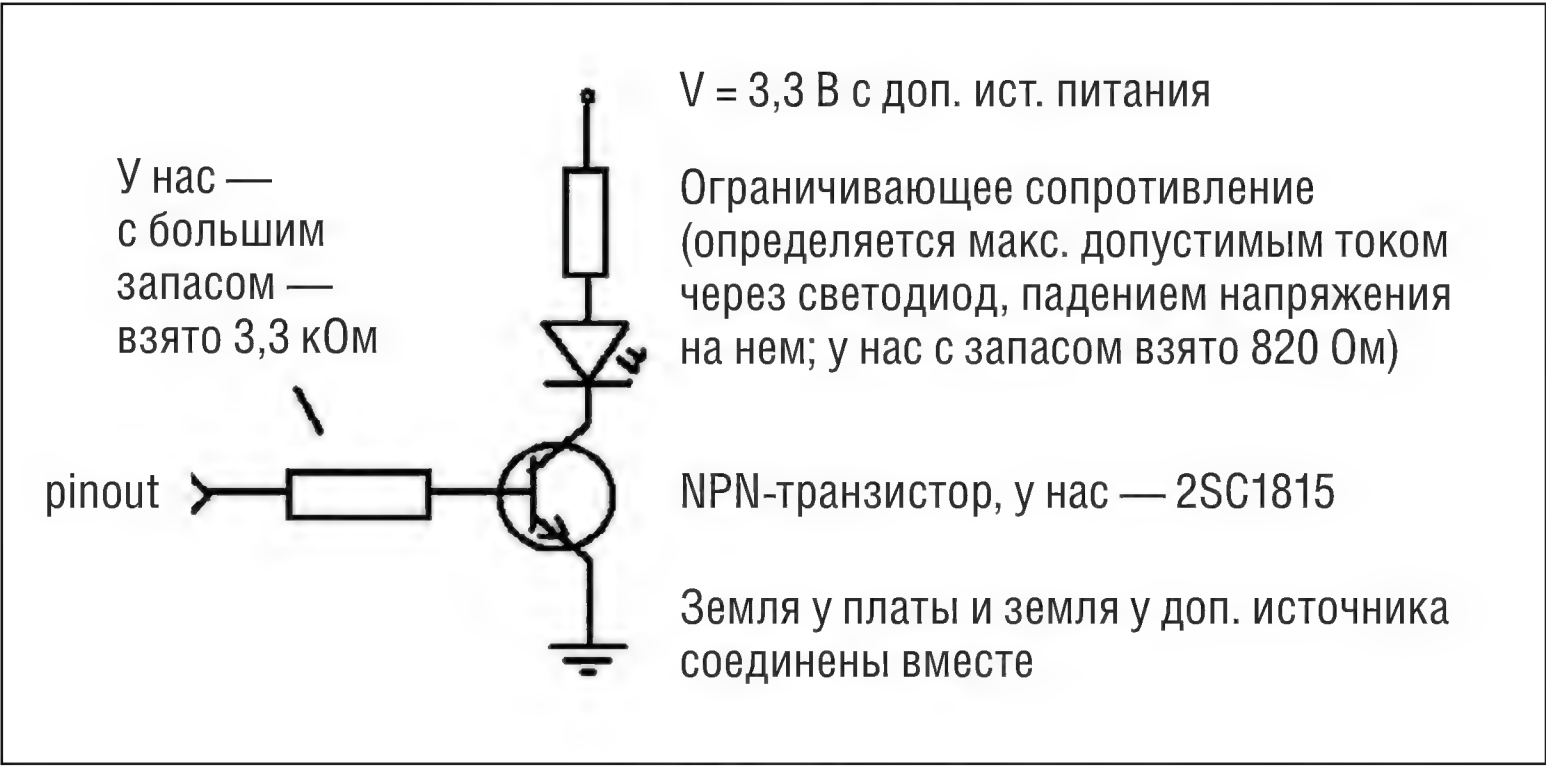


Рис. 3. Сборка «прототипа гирлянды» — пока только с одним светодиодом.

Схема ключа показана на рис. 2. В качестве NPN-транзистора был использован 2SC1815Y-TOS. При этом нужно учитывать допустимый ток, снимаемый с его коллектора. Для 2SC1815Y-TOS это 100 мА. В нашем случае нагрузка была существенно меньше, и получился солидный запас прочности. Если нагрузка близка к (или тем более превышает) 100 мА, то необходимо подобрать другой транзистор или внести иную модификацию в схему.

Последние штрихи

Используем следующий файл ограничений:

```
NET "clk" LOC = "P89" | IOSTANDARD = LVCMOS25 | PERIOD = 31.25ns ;
NET p15 LOC = "P15" | IOSTANDARD=LVTTL;
NET p16 LOC = "P16" | IOSTANDARD=LVTTL;
NET p17 LOC = "P17" | IOSTANDARD=LVTTL;
NET "tx" LOC="P90" | IOSTANDARD=LVTTL | DRIVE=8 | SLEW=FAST;
NET "rx" LOC="P88" | IOSTANDARD=LVTTL | DRIVE=8 | SLEW=FAST | PULLUP;
```

Если потребуется подсоединить больше выводов, также необходимо добавить для них строки в этот файл. Для этого ещё потребуется подправить фрагмент управляющего модуля, где реализуется команда вывода в порт (icode_out), а в файле сборки инициализировать порт вывода.

Заключение

На нашем уроке было показано, как построить очень простой процессор. Его функциональность весьма ограничена. Отталкиваясь от полученной реализации, можно создать более функциональные и более производительные процессоры. В частности, можно увеличить объём оперативной памяти, ввести систему прерываний и добавить таймеры, watchdog и иные функциональные блоки, изменить набор инструкций, сделав его выразительнее и удобнее, перейти к более эффективному двоичному коду и написать утилиту ассемблирования. LXF

LINUX FORMAT

Подписывайтесь и читайте Linux Format на iPad или iPhone!

Доступно
в AppStore



А если у вас Android, подпишитесь
на Linux Format через Zinio!

 **zinio**
Доступно в Google Play



Cython: Python разгоняется

Джонни Бидвелл показывает, как встроить в вашу систему ускоритель Cython, взяв в качестве рабочего примера сжатие изображений.



Наш эксперт

Джонни Бидвелл горой стоит за то, чтобы все делалось вовремя.



Python — потрясающий язык. У него понятный и простой в изучении синтаксис, и вы можете сделать очень многое всего несколькими строками. Но он отнюдь не быстр, что иногда способно привести к «отказу от сотрудничества». Основная причина в том, что Python — язык интерпретируемый: он считывается строка за строкой и на ходу конвертируется в промежуточный байт-код, перетряхиваемый и исполняемый в CPU. Это требует времени, но облегчает жизнь: незачем перекомпилировать свой код при каждой правке, и нет нужды вводить переменные.

Интерпретатор сообразит, каким типом данных все будет, и даже если вы замените, скажем, список на целочисленную переменную, он без единой жалобы употребит ваши изменения. Если вы на самом деле хотите, чтобы ваш код Python ускорился, перепишите его на C, и быстрота обеспечена. Однако это легче сказать, чем сделать: C — язык непростой, а вам частенько бывает нужно всего лишь расшить несколько узких мест в коде.

Познакомьтесь с Cython, который неверно воспринимают как переводчик Python-в-C. Отчасти это верно: Cython возьмет ваш код Python (слегка измененный) и выдаст файл на C, который вы сможете скомпилировать и затем импортировать как модуль расширения с ускоренными версиями всех функций в вашем первичном коде. Однако вам все равно нужен ваш первичный код:

полученный модуль присутствует только для того, чтобы конвертировать соответствующие его части в родной машинный код вместо байт-кода Python.

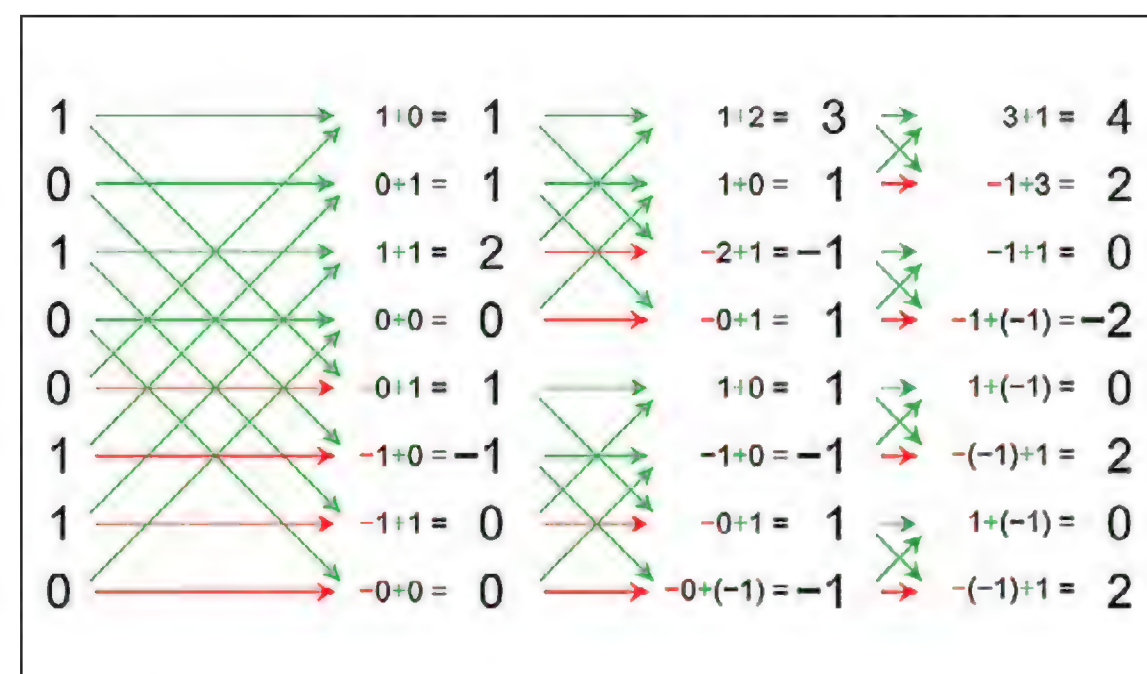
Язык Cython — это надмножество Python, поэтому (за исключением нескольких специализированных модулей и функций) любой действующий Python — это также действующий Cython, и как таковой может быть сохранен в виде файла PYX и передан в бинарник Cython. Однако для оптимальной «cython'изации» нужно использовать дополнительные ключевые слова Cython, которые могут создавать типы переменных (включая параметры функций и типы возврата) и предоставлять более быстрый доступ к массивам.

Многие программы на самом деле от подобного обращения с Cython ничего не выиграют, и, не проявив бдительности, вы можете в конечном итоге все затормозить. Например, если ваша программа тратит львиную долю времени на рисование графиков или сильно зависит от I/O, тут вам Cython не поможет. Однако если большую часть своей жизни ваша программа проводит, петляя по массивам, перемещая туда и сюда биты и занимаясь вычислениями, то вам повезло.

Сжатие данных

Мы применим Cython для ускорения черновой реализации быстрого преобразования Уолша–Адамара [Fast Walsh–Hadamard Transform]. Мы будем использовать это преобразование, чтобы сжимать (с потерями) изображения в шкале оттенков серого, хотя данный принцип применим к любому типу данных. На заре развития спутниковой фотографии подобные техники применяло НАСА, поскольку преобразование использует только незатратные по времени действия сложения и вычитания, и благодаря парочке математических хитростей эти действия можно сэкономить (до $O(n \log n)$ вместо $O(n^2)$, если кому интересно).

8-битное изображение в оттенках серого можно представить в виде списка целочисленных переменных без знака, от 0 до 255 — например, байтов. Интенсивности каждого пикселя



➤ Сложения и вычитания с замещением рассчитывают спектр Уолша без необходимости умножения на большую матрицу.

Скорая помощь

Если вы уже ис-пробовали `cdef`, и вам все равно не хватает скорости, можете передать директивы компилятору (типа бесславной `-O3`). Загляните в официальную документацию — <http://bit.ly/CythonDocs>.

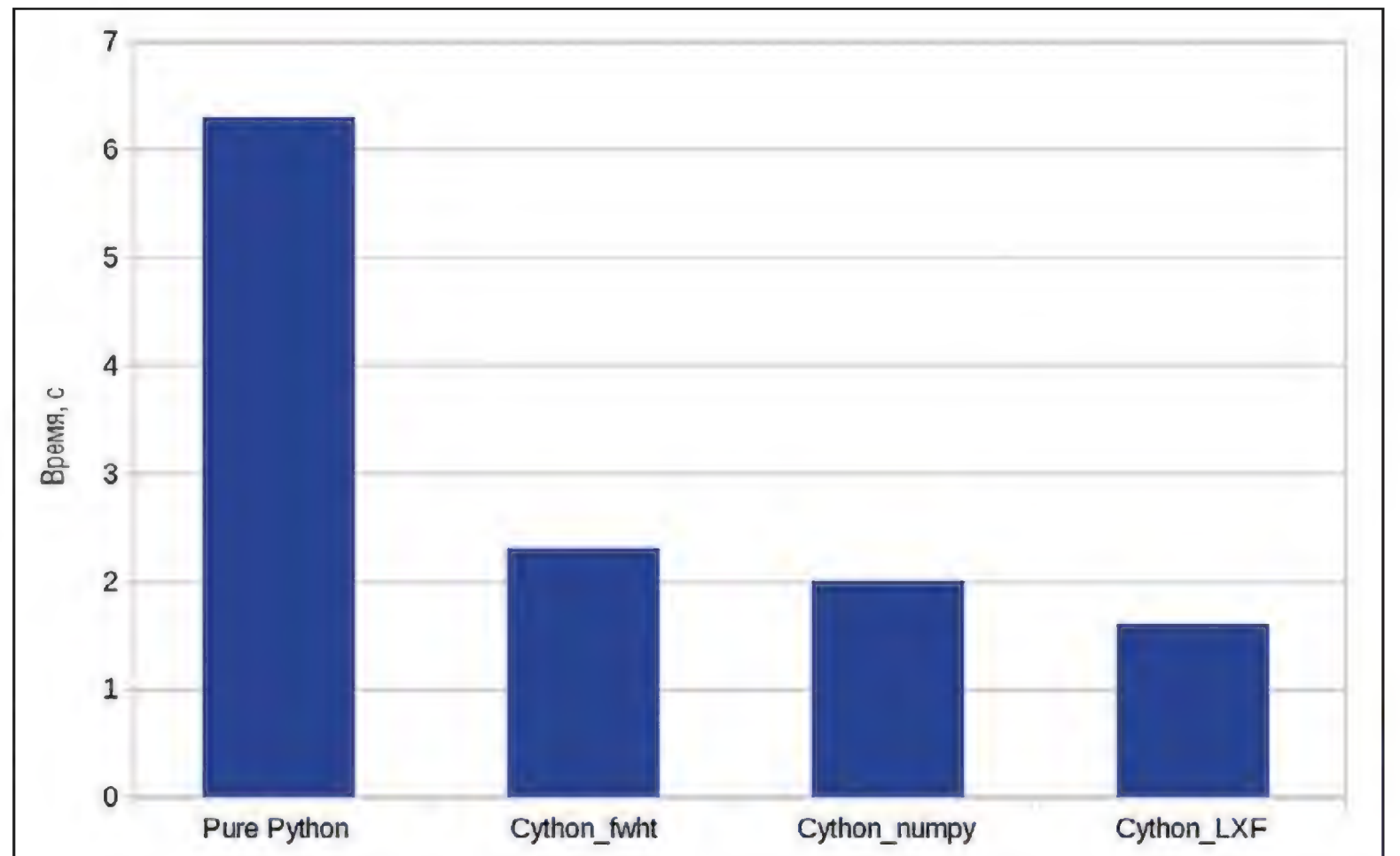
соответствует один байт, так что изображение 256×256 будет занимать 65536 байт или 64 КБ. Функции Уолша — хорошо известное семейство функций, которые принимают значения 1 или -1. Суммируя компоненты, выражаемые функциями Уолша, можно представить любую функцию с дискретным значением. Например, ряд пикселей в нашем изображении или даже все изображение целиком можно точно воспроизвести, скажем, суммировав одну функцию Уолша 300 раз, 84 раза вычтя другую, прибавив еще 6 штук третьей и 2 раза вычтя четвертую. Преобразование Уолша–Адамара быстро и эффективно сообщит вам, какие именно коэффициенты каким функциям отвечают.

На практике, если только вы не работаете с придуманными данными, никакой пользы в хранении родительской функции подобным образом нет (обычно нужно суммировать столько функций, сколько у вас пикселей или пунктов данных). Однако если вас не особо беспокоит утрата данных, то вы можете получить весьма неплохое приближенное значение своих данных, отбрасывая функции Уолша с младшими коэффициентами. Особо беспокоиться по поводу сохранения или способов сохранения приблизительного изображения не будем. Вместо этого мы сделаем обоснованное предположение по поводу размера файла — в частности, что каждый коэффициент занимает 10 битов хранения (чтобы он мог принять значения от -511 до 511) в дополнение к нескольким битам для каждого индекса. Мы можем показать, как выглядит сжатое изображение, однако в Python оно по-прежнему будет представлено несжатым массивом.

Преобразование Уолша–Адамара

Преобразование Уолша–Адамара обычно представляется как матричное преобразование, где квадратная матрица размерности степени двойки умножается на вектор-столбец длиной степени двойки (наши данные). Матрица ортогональная и (если подобрать должный масштабный коэффициент) унитарная, то есть можно вернуть преобразованный вектор в исходный, применив преобразование снова. Это пример матрицы Адамара: элементы матрицы (с точностью до масштабного коэффициента) все являются ± 1 , а строки образуют функции Уолша. (Забавный факт: это было открыто за 20 лет до рождения Уолша, в контексте устранения перекрестных помех в параллельных телеграфных линиях).

Быстрое преобразование Уолша–Адамара задействует рекурсивную структуру матрицы Уолша (которую можно определить как тензорное произведение матриц 2×2) для ускорения расчетов, используя локальные вычисления, приведенные в диаграмме на следующей странице.



В приведенном ниже коде мы немного схитрим и используем функцию `log2` из NumPy. Не стоит особо переживать по поводу логической загадки, связанной с `j` и `k` ниже. Это просто красивый способ воссоздать структуру «мотылька» на диаграмме. Алгоритм работает с вводимой информацией, складывая и вычитая пары элементов и записывая результат прямо туда же, поэтому ему не нужно возвращать что-либо в виде результата:

```
import numpy as np
def fwht(arr):
    n = len(arr)
    b = int(np.log2(n))
    for bit in range(b):
        for k in range(n):
            if k & (1 << bit) == 0:
                j = (1 << bit) | k
                tmp = arr[k]
                arr[k] += arr[j]
                arr[j] = tmp - arr[j]
```

Побитовые операторы `<<` и `>>` в Python не слишком поворотливы, но в C соответствуют машинному уровню работы и работают намного быстрее, чем эквивалентное настоящее умножение или деление целочисленных переменных на степень двойки.

Наш алгоритм сжатия считывает, используя библиотеку изображений Python, изображение в оттенках серого в виде одномерного массива. Мы поделим наш массив на фрагменты и выполним преобразование над этими фрагментами. Нам нужна

» Первые оптимизации приносят самую большую пользу. После этого можно потратить часы на попытки сэкономить миллисекунды.

»

В декорациях Cython

Помимо типа переменных, мы также можем указать типы входов и возврата для функций. Для этого мы определяем функцию с помощью `cdef` и затем указываем ее тип возврата перед ее именем. Например, наша основная функция `fwht` ничего не возвращает, и, следовательно, ее тип — `void`. После того, как мы оптимизировали начинку из `fwht`, она применяется в качестве ввода просмотра памяти для целых (`int`) C, поэтому определяется как

```
cdef void fwht(int[:] arr)
```

Использование `cdef` означает, что ваша функция не будет доступна для других модулей Python, но если вам нужно также, чтобы функция работала

при доступе извне, можете использовать `cpdef` (что повлечет некоторые издержки). Применив `cimport` к модулю `cython`, мы можем получить доступ к нескольким декораторам, которые меняют поведение на уровне функции. Например, чтобы отключить профайлинг для отдельной функции, используйте

```
@cython.profile(False)
def too_cool_for_timing:
```

Вы поймете, что это особенно удобно при использовании в связке с ключевым словом `inline`, которое используется для «разворачивания» небольших, но часто используемых функций и для

сокращения издержек, связанных с вызовом функции. Вам нужно поставить ключевое слово `inline` сразу после `cdef`.

И, наконец, есть пара довольно популярных «опасных» вещей, а именно:

```
@cython.boundscheck(False)
```

и

```
@cython.cdivision(True)
```

которые соответственно деактивируют проверку за пределами массива и проверку деления на ноль. Прежде чем это делать, убедитесь, что ваш код правильный, поскольку тут возникает потенциальный риск разрушить память.

» Не хотите пропустить номер? Подпишитесь на www.linuxformat.ru/subscribe!

функция для определения и хранения наибольших коэффициентов, полученных от каждого из них. Есть смысл также выполнить здесь сдвиг и округление; вы можете увидеть результат в функции **squishChunk()** в файлах **LXFDVD**. Декомпрессия через функцию **expandImage()** включает работу с каждым фрагментом по подборке индексов коэффициентов и модулей в вектор и затем по выполнению преобразования снова и возврату всего в диапазон 0–255. Затем фрагменты снова объединяются, и мы используем метод **show()** для отображения полученного в результате сжатого изображения с возможными потерями. Этот метод требует работы с X-сервером, поскольку использует для отображения программу **xv** (которая вам тоже понадобится).

Вы можете проверить, все ли работает, скопировав директорию с **LXFDVD** в локальную папку и запустив оттуда

```
$ python proftest.py
```

Эта программа сожмет, а потом расширит фотографию из нового обиталища модуля **Philae**: вы увидите ее на соседней странице. Можете поэкспериментировать с параметрами **chunksize** и **nterms** в начале файла **fwht_python**.

Исходные величины (32 и 8) дают номинальное соотношение сжатия более 2:1, хотя это бессмысленно, если мы не храним сжатых данных. Это также можно солидно улучшить, изменяя количество компонентов для каждого фрагмента — области одного цвета требуют только одного компонента. Мы можем создать профиль этого кода с помощью модуля **cProfile**, запустив

```
$ python -m cProfile proftest.py
```

Он перечислит все функции, используемые в программе, включая все странные функции для декодирования изображения PNG, чтобы мы смогли отфильтровать все это и увидеть свои действия, добавив к вышеупомянутому **grep fwht**. На пыльном компьютере в офисе **LXF** все исполнение заняло около пяти секунд, из которых более трех ушло на функцию **fwht**: оно и понятно, ведь **fwht** — сердцевина нашей программы.

Нам может удасться немного повысить скорость, используя вместо списков массивы **NumPy**. Их можно инициализировать нулями, но чуть быстрее будет пропустить этот шаг (тогда массив будет содержать случайные данные, сохранившиеся в отведенной ему памяти), если вы уверены, что позднее параметры будут введены. Вам также нужно указать тип данных для массива, и лучше не использовать методы для списков Python, вроде **len()**. Код вы найдете в файле **fwht_numpy.py**. Кроме того, вы обнаружите, что на его выполнение уходит в два раза больше времени — сопоставительный анализ полон сюрпризов. Несмотря на это мелкое разочарование, мы продолжаем использовать наши массивы — с ними **Cython** может выполнять работу лучше.

Войдите в Cython

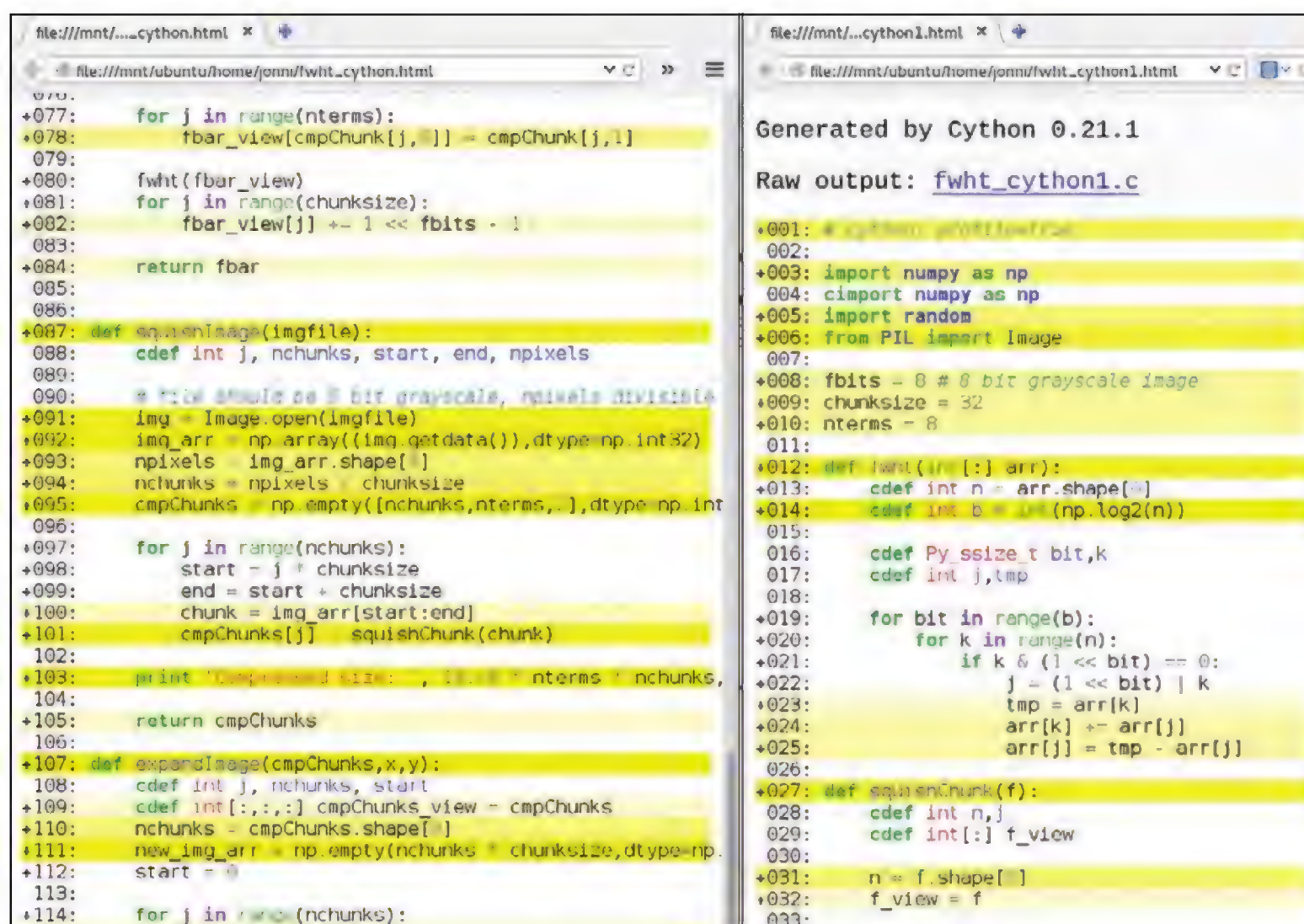
Имеет смысл сначала сосредоточить усилия на ускорении функции **fwht()**, которая на данный момент вполне читаема. Простым первым шагом будет указать типы данных для всех локальных переменных в этой функции. Хотя все они являются целочисленными переменными и вы можете обойтись объявлением их как таковых, для индексов цикла **for** есть специальный тип **Py_ssize_t**, так что мы можем воспользоваться им. Добавьте следующие строки в начале функции **fwht** в **fwht_numpy.py** и сохраните файл, скажем, как **fwht_cython1.pyx**:

```
def fwht(arr):
    cdef int n = arr.shape[0]
    cdef int b = int(np.log2(n))
    cdef Py_ssize_t bit, k
    cdef int j, tmp

    Теперь скомандуйте
    $ cython -a fwht_cython1.pyx[/b]
```

Это создаст довольно сумбурный **C** в файле под названием **fwht_cython1.c**. Параметр **-a** дает **Cython** команду дополнительно создать с тем же названием файл **HTML**, на который вам следует взглянуть. Вновь введенные вами строки — белого цвета, а большая часть остального кода — разных оттенков желтого. Вы даже можете щелкнуть по каждой строке, чтобы посмотреть, как она выглядит в **C**; при этом вы обнаружите, что желтые

➤ Использование опции **-a** создает файлы **HTML**, показывающие вам незапятнанно белый **C**-образный код и грязно-желтый код **Python**.



Сравнительный анализ

Очень легко начать относиться к сравнительному анализу слишком серьезно — энтузиасты видеокарт имеют в этом немалый опыт. В нашем руководстве мы используем модуль **cProfile**, который позволяет посчитать и выяснить время каждого вызова функции. Это предоставит ценные данные о проблемных местах в вашем коде, вначале неочевидных.

CProfile разработан с целью быть как можно более легковесным и незаметным, но если у вас есть крошечная функция, вызываемая миллионы раз, то это означает миллионы вызовов, и каждый

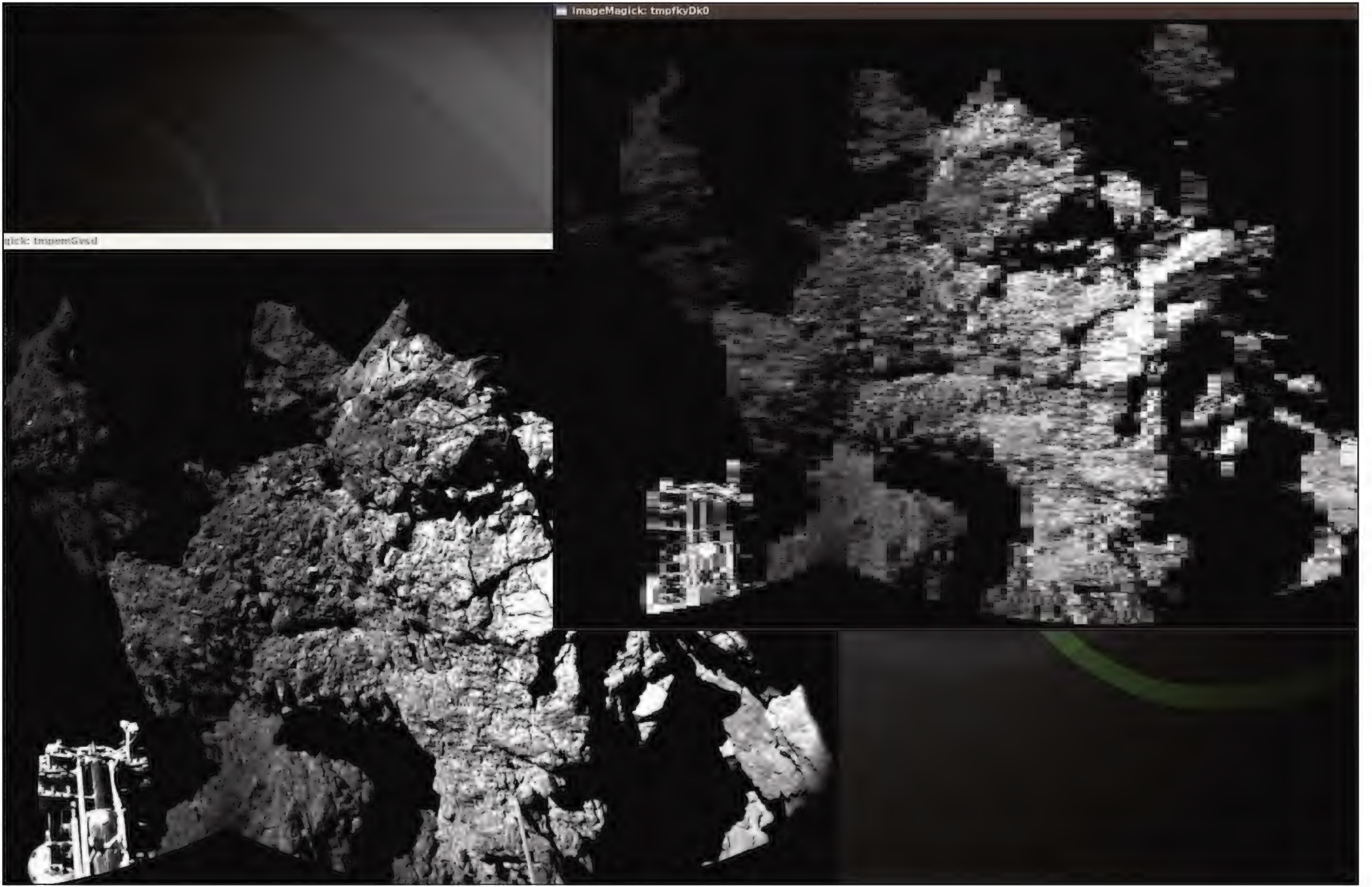
из них добавляет нагрузку. Если функция реально мала, то на сравнительный анализ уходит больше времени, чем на действия функции, поэтому результат довольно бесполезен. Если вы уверены, что увеличить скорость работы небольшой функции уже невозможно, отключите профайлер. Если же подобной уверенности у вас нет, продолжайте сравнительный анализ подобных небольших функций, но не сомневайтесь, что они будут работать намного быстрее, если не будет мешать профайлер.

Если вы хотите измерить «фактическое время», то есть общее время на запуск фрагмента кода, вам

больше подойдет модуль **timeit**. Например, чтобы проверить (из интерпретатора) функцию под названием **testfunction()** из модуля **testmodule** три раза:

```
import timeit
timeit.timeit(stmt='testmodule.testfunction()',
    setup='import testmodule', number=3)
```

Величина по умолчанию для **number** — миллион, поэтому мы и указали здесь собственное значение. Вам нужно указать свой модуль в параметре **setup**, даже если вы ранее уже импортировали его, поскольку **timeit** не унаследует этого пространства имен.



строки соответствуют более длинному или более сложному коду. Нам все же нужно сделать сопоставительный анализ, поэтому добавьте следующий декоратор в верхней части файла:

```
# cython: profile=True

Компиляция вашего кода Cython требует некоторых усилий. Вы можете сделать ее вручную, но проще использовать функцию cythonize и модуль distutils. Создайте файл setup.py (пример вы найдете на LXFDVD) со следующим содержимым:
```

```
from distutils.core import setup
from distutils.extension import Extension
import cythonize

setup(
    ext_modules = cythonize(["fwht_cython1.pyx"])
)
```

Теперь, если вы запустите `$ python setup.py build_ext --inplace` все скомпилируется, и вы можете изменить **proftest.py**, чтобы использовать свой новый модуль **fwht_cython1**. Его сопоставительный тест на нашей машине (по-прежнему с массивами NumPy) замедлил работу — только выполнение функции **fwht()** заняло 20 секунд. Но не печальтесь: причина замедления в том, что C должен получить доступ к **arr** через методы Python и NumPy.

Просмотры памяти

Недавно Cython представил новый способ доступа к данным массивов через так называемые просмотры памяти [memory views]. Они позволяют C получить прямой доступ к данным массива в памяти, и, следовательно, чертовски быстры. Но они не работают со списками Python, потому мы и придерживаемся наших более медленных массивов NumPy. Мы заставим **fwht()** выполнять просмотр памяти, изменив строку определения:

```
def fwht(int[:] arr):
    Функции squishChunk() и expandChunk() тоже нужно будет изменить. Итак, продолжим и определим f_view в squishChunk() следующим образом:
```

```
cdef int[:] f_view
f_view = f

Замените все дальнейшие ссылки в функции на массив f на f_view, за исключением вызова enumerate, который является функцией Python. Подобным же образом определите fbar_view в expandChunk(), и замените все ссылки fbar, кроме оператора return fbar. Пока мы этим занимаемся, мы можем с таким же успехом придать всем j и n и иже с ними тип int. Затем заново запустите setup.py и проведите анализ. Теперь мы кашеварим уже не на дровах, а на газе — общее время выполнения занимает менее трех секунд, и большая его часть тратится на функцию squishChunk(). Узким местом здесь будет ранжирование наших коэффициентов и прав, поэтому давайте выделим их в отдельную функцию rankArray(), которая не так сильно зависит от конструкций Python:
```

```
def rankArray(int[:] F):
    cdef int n, j
    n = F.shape[0]
    Franked = np.empty([n,3], dtype=np.int32)
    cdef int[:,:] Franked_view = Franked
    for j in range(n):
        Franked_view[j,0] = j
        Franked_view[j,1] = F[j]
        Franked_view[j,2] = -abs(F[j])
    Franked = Franked[Franked[:,2].argsort()]
    return Franked[:nterms,:2]
```

Теперь измените строку **return** в **squishChunk()**, чтобы использовать это, и проверьте анализ. На нашем компьютере нам удалось выкроить почти секунду, и мы были счастливы. Далее наши успехи стали более скромными, но в файле **fwht_cython.pyx** на **LXFDVD** мы обеспечили максимально возможную оптимизацию. Общее время выполнения снизилось до 1,6 секунды — попробуйте-ка сделать лучше! **LXF**

➤ Открытка с кометы 67P в оригинале (слева) и в сильно сжатом виде (справа). Посадка на внеземных объектах — та редкая удача, которая только и может заставить суровых космических инженеров обняться.

R: Приступим

Михалис Цукалос вводит нас в язык программирования для статистики и исследует его приложение к анализу данных мониторинга системы.



Наш эксперт

Михалис Цукалос обожает статистику и мониторинг данных. Мало того, что он математик — он еще и администратор Unix, программист и администратор базы данных.

R — это проект GNU, основанный на S, языке для статистических расчетов, и среде, разработанной в знаменитой Лаборатории Белла [Bell Labs]. В принципе, можно считать R свободной версией языка S. Система R поддерживает ряд статистических процедур, включая линейные и обобщенные линейные модели, модели нелинейной регрессии, анализ на основе временного ряда, классические параметрические и непараметрические критерии, выделение в кластеры и сглаживание.

С целью повсеместной применимости, мы рассмотрим в этом руководстве только версию R командной строки, однако вам не стоит пугаться этого языка, даже если вы не слишком дружите с математикой, поскольку вы можете воспользоваться имеющимся GUI. Самый популярный из них (и мой личный любимец) — RStudio (www.rstudio.com).

Установка R

В системе на базе Debian 7 можно установить R, введя

```
# apt-get install r-base
```

Затем можете запустить R и перейти в его оболочку, просто введя R в оболочке Unix. Следующий результат показывает, насколько легко делать вычисления в R:

```
> 4 + 5
[1] 9
> 4 * 4
[1] 16
> 4 ^ 3
[1] 64
> 4 ^ 10
[1] 1048576
> 4 / 3
[1] 1.333333
```

R умеет автоматически считывать данные из структурированных текстовых файлов с помощью команды **read.table()**. Самая полезная команда R для получения полного обзора набора данных — **summary()**:

```
> data <- read.table("uptime.data", header=TRUE)
> summary(data)
```

	X1min	X5min	X15min
Min. :	0.00000	0.01000	0.05000
1st Qu.:	0.00000	0.01000	0.05000
Median :	0.00000	0.01000	0.05000
Mean :	0.02028	0.02491	0.05553
3rd Qu.:	0.00000	0.02000	0.05000
Max. :	2.47000	2.15000	1.05000

Подробности о команде **read.table()** выдаст команда **help(read.table)**.

Создание новых функций R

Сначала давайте рассмотрим код R, необходимый для реализации двух новых функций: одной — для поиска чисел Фибоначчи, а другой для поиска факториала целого числа. При определении собственных функций убедитесь, что у них уникальные имена. Код для определения чисел Фибоначчи следующий:

```
myFibo = function(i)
{
```

```
  if ( i == 0)
  {
    return(0)
  }
  if ( i == 1)
  {
    return(1)
  }
  if ( i == 2)
  {
    return(1)
  }
  return (myFibo(i-1) + myFibo(i-2))
}
```

Код должен быть вам знаком. Как видите, инициализировать или объявлять переменные не надо. Однако будьте аккуратнее, потому что из-за этого иногда возникают ошибки или иные неприятности.

Сохранив код, можете загрузить его с помощью **source()**, при условии, что вашей рабочей директорией является директория, где размещается **fibonacci.R** (в ином случае введите весь путь):

```
> source("fibonacci.R")
> myFibo(4)
[1] 3
> myFibo(15)
[1] 610
> myFibo(26)
[1] 121393
```

Если все в порядке, то по выполнению команды **source()** R не выведет никакой информации. Видно также, что R автоматически печатает результат — значения функции. В случае ошибки в вашем коде R, R выведет для вас удобное сообщение об ошибке:

```
> source("fibonacci.R")
Error in source("fibonacci.R") :
[Ошибка в источнике («fibonacci.R»)]
fibonacci.R:5:16: unexpected numeric constant
[неуместная числовая константа]
4: {
5: return 0
^
```

Код R, пригодный для поиска факториала целого числа, таков:

```
myFactorial = function(i)
{
  if ( i == 0 )
  {
    return(1)
  }
  if ( i < 0 )
  {
    return(-1)
  }
  result = 1
  for ( k in 1:i )
  {
    result = k*result
  }
}
```

Скорая помощь



Азбука теории вероятности: бросая шестигранную игральную кость, вы угадываете правильное число с вероятностью 1/6. При двух бросаниях вероятность отгадки сократится до (1/6)*(1/6), то есть 1/36.

Определения статистики

Следующие определения из статистики помогут вам разобраться в результатах команды **summary()**:

- » **Min** Минимальное число в наборе данных.
- » **Median** Медиана — элемент, который делит упорядоченный набор данных на два поднабора (правый и левый) с одинаковым количеством элементов.
- » **1st Qu** Первая квартиль (Q_1) — это значение, характеризующееся тем, что 25% значений данных

в наборе данных меньше, чем Q_1 , а 75% — больше. По-простому, можно считать Q_1 медианой левого поднабора упорядоченного набора данных. Учтите, что сама по себе Q_1 не обязательно является значением, имеющимся в наборе данных.

- » **Mean** Среднее — сумма всех элементов набора данных, деленная на их количество.
- » **3rd Qu** Третья квартиль (Q_3) — такое значение, что 75% данных в наборе данных меньше, чем Q_3 ,

а 25% — больше. То есть медиана правого поднабора обработанного набора данных. И снова учтите, что Q_3 не обязательно принадлежит к набору данных.

Помните также, что квартили высчитываются по-разному: общепринятого наилучшего способа нет. В другом статистическом пакете Q_1 и Q_3 могут получиться другими.

- » **Max** Максимальное число в наборе данных.

```
}
return(result)
}
```

На сей раз реализация немного другая: вместо рекурсии она использует цикл **for**. По сравнению с другими языками программирования синтаксис цикла **for** немного отличается, но его легко понять и запомнить.

Создание собственных пакетов

Теперь давайте упорядочим код, поместив его в пакет R: пакеты — хороший способ упорядочения. Функции в пакетах могут иметь любое имя, лишь бы оно было уникально: от дублирования имен функций проку мало. Если они дублируются в разных пакетах, при вызове функции перед ее именем добавляйте имя пакета — например, **LinuxFormat::function()** вместо просто **function()**.

Вот что надо сделать, чтобы создать пакет R, который мы хотим назвать **LinuxFormat** и который будет содержать функции **myFibo()** и **myFactorial()**:

```
> ls()
character(0)
> source("# # нажатие Tab
fibonacci.R factorial.R sort.R
> source("factorial.R")
> source("fibonacci.R")
> ls()
[1] "myFactorial" "myFibo"
> package.skeleton("LinuxFormat")
Creating directories ...
[Создание каталогов]
Creating DESCRIPTION ...
[Создание описания]
Creating NAMESPACE ...
[Создание пространства имен]
Creating Read-and-delete-me ...
[Создание файла Read-and-delete-me]
Saving functions and data ...
[Сохранение функций и данных]
Making help files ...
[Создание файлов справки]
Done.
[Готово]
```

Дальнейшие шаги будут описаны в файле **./LinuxFormat/Read-and-delete-me**. Последняя команда R создает новую директорию с именем **LinuxFormat** — тем же, что и у пакета R — и вы можете тщательно изучить ее содержимое с помощью:

```
$ ls -lR LinuxFormat/
LinuxFormat/:
total 20
```

```
-rw-r--r-- 1 mtsouk mtsouk 284 Nov 4 10:18 DESCRIPTION
drwxr-xr-x 2 mtsouk mtsouk 4096 Nov 4 10:18 man
-rw-r--r-- 1 mtsouk mtsouk 31 Nov 4 10:18 NAMESPACE
...
```

Пакет **LinuxFormat** автоматически будет содержать две функции, поскольку вызовов **source()** было два. Чтобы ваши пакеты были доступны всем в вашей системе Linux, установите их от имени **root**:

```
# R CMD INSTALL LinuxFormat
* installing to library '/usr/local/lib/R/site-library'
[установка в библиотеку]
* installing *source* package 'LinuxFormat' ...
[установка *исходного* пакета]
** R
...
* DONE (LinuxFormat)
[ГОТОВО]
```

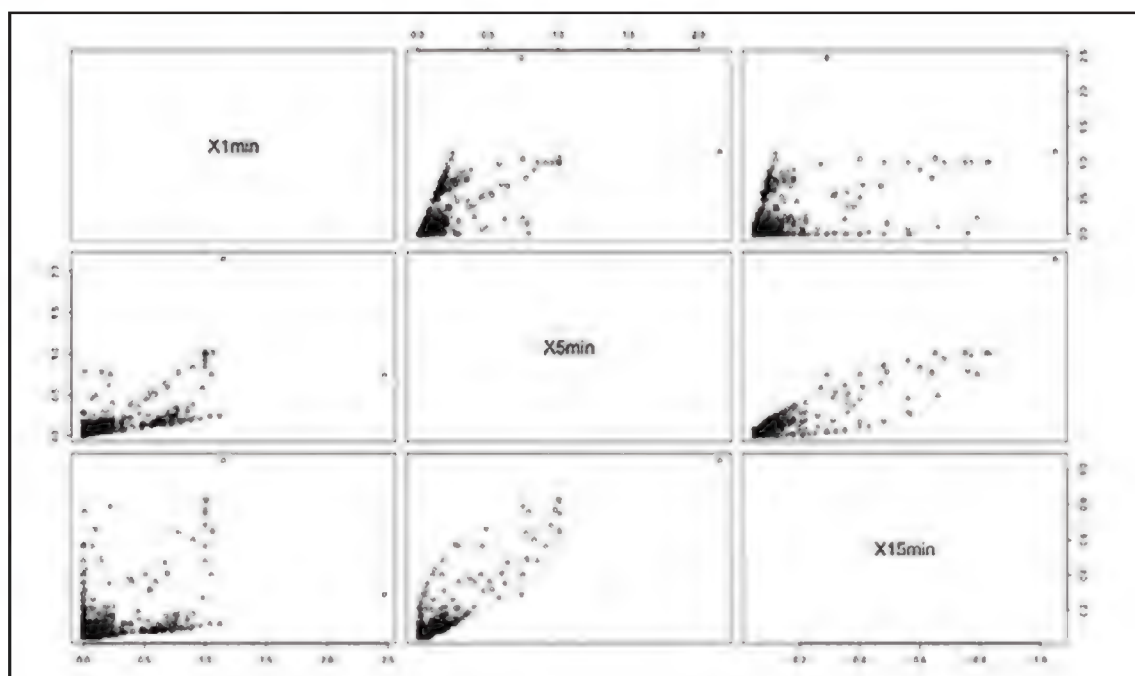
Следующий код и его результаты доказывают, что пакет был успешно установлен:

```
# ls -l /usr/local/lib/R/site-library
total 4
drwxr-xr-x 6 root staff 4096 Nov 4 10:33 LinuxFormat
```

Предупреждение: перед установкой пакета вы должны отредактировать файлы **myFactorial.Rd** и **myFibo.Rd** и заполнить поля **\title**. Не сделав этого, вы получите сообщение об ошибке, и установка не выполнится.

Далее вы можете использовать новый пакет следующим образом:

```
> require(LinuxFormat)
Loading required package: LinuxFormat
> ls(getNamespace("LinuxFormat"))
[1] "myFactorial" "myFibo"
> ls()
character(0)
> myFibo(12)
```



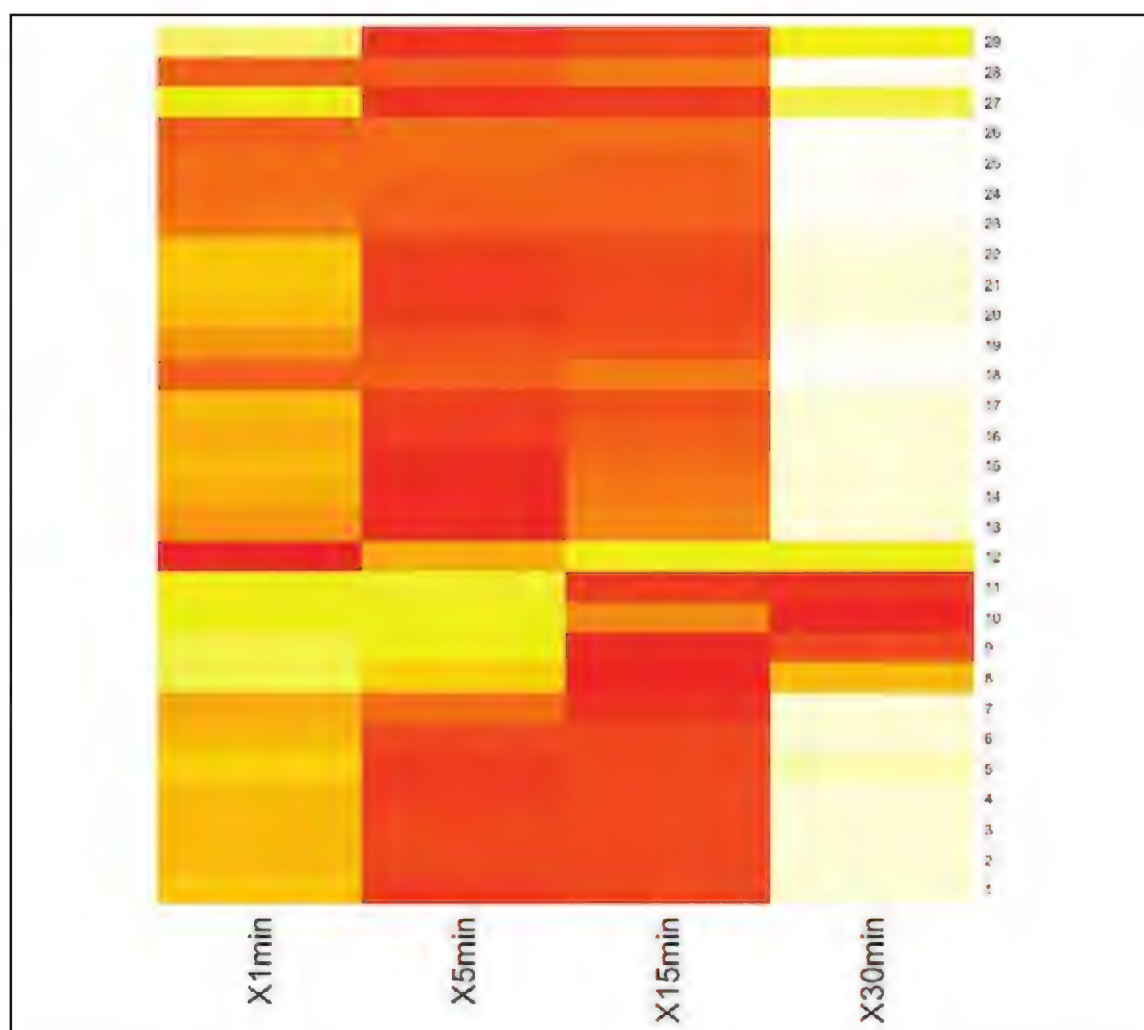
В развитие Азбуки теории вероятности (см. стр. 92): при одновременном бросании двух костей вероятность угадать оба числа будет другой, поскольку не надо угадывать порядок появления значений.



» Результат команды **pairs()** (со следующей страницы). Пакет R **ggplot2** использует **ggpairs()**, это улучшает вывод.

» **Не хотите пропустить номер?** Подпишитесь на www.linuxformat.ru/subscribe!

» **Теплокарта** — хороший и визуально эффектный способ представления данных. Теплокарты отлично подходят для анализа данных мониторинга по латентности и загрузке.



```
[1] 144
> LinuxFormat::myFibo(12)
[1] 144
```

Теперь рассмотрим, как употребить R для анализа реальных данных. Я буду анализировать собранный мной набор данных мониторинга системы, но вы можете взять собственный набор данных.

Команда **pairs()** предлагает очень удобный способ поиска отношений между переменными. Если вы используете **ggplot2** — мощный пакет R для создания графиков, который сам по себе достоин отдельной статьи — вы можете также использовать **ggpairs()**, улучшенную версию **pairs()**, которая рассчитывает и добавляет в результат коэффициент корреляции. Это статистический термин, применяемый для описания зависимости между двумя переменными. Проще говоря, чем ближе значение коэффициента корреляции к 0, тем слабее связь между двумя переменными — т.е. они ближе к некоррелированности. Чем ближе значение коэффициента к +1 или -1, тем сильнее корреляция между переменными. Положительный коэффициент корреляции указывает, что при увеличении одной переменной другая переменная тоже увеличится. Отрицательный — что при увеличении одной переменной другая будет уменьшаться.

Результаты от **pairs()** и **ggpairs()** генерируются следующими командами:

```
> data <- read.table("uptime.data", header=TRUE)
> pairs(data)
> require(ggplot2)
> require(GGally)
> require(CCA)
> ggpairs(data)
```

На рисунке стр. 93 показан результат команды **pairs()**, примененной к моим данным мониторинга системы. Видно, что переменные X5min и X15min «более связаны», чем переменные X1min и X15min. Иными словами, средние значения нагрузки системы Linux претерпевают более резкие изменения за 1 минуту, чем за пять или пятнадцать минут.

Сохраним графический результат, выданный R, в новом файле под названием **filename.png**. Сначала нужно открыть «устройство» с помощью **png()**, **bmp()** или **pdf()**; затем вывести графическое изображение всего, чего хотите, с помощью выбранных вами команд. И, наконец, при удаленном использовании R, надо закрыть устройство — учтите, что в скриптах R эта конечная команда не обязательна:

```
png(filename="filename.png")
# Далее выполняются ваши команды вывода
dev.off()
```

Теперь давайте создадим столбчатую диаграмму — это отличный способ наглядно показать распределение, разброс и медиану набора данных. Верх и низ диаграммы — это первая и третья квартили набора данных, а горизонтальная линия посередине — медиана. «Усы» над и под столбиком показывают разброс вне квартилей, а круги над или под усами показывают значения данных-выбросов.

Столбчатая диаграмма превосходно визуализирует показатели, например, времени, проведенном на странице посетителем, и времени обслуживания страницы. Например, диаграмма внизу использует множественную выборку трех средних значений нагрузки, взятых из команды **uptime**. Она была создана тремя командами R:

```
> data <- read.table("uptime.data", header=TRUE)
> boxplot(data, ylab="Время работы", xlab="Значения выборки",
col="lightblue", border="blue", main="Графики средней нагрузки")
> grid()
```

Первая команда считывает данные из внешнего файла и сохраняет их в новую переменную с именем **data**. Вторая команда создает из набора значений **data** столбчатую диаграмму. Последняя команда рисует сетку на экране, чтобы сделать вывод красивее.

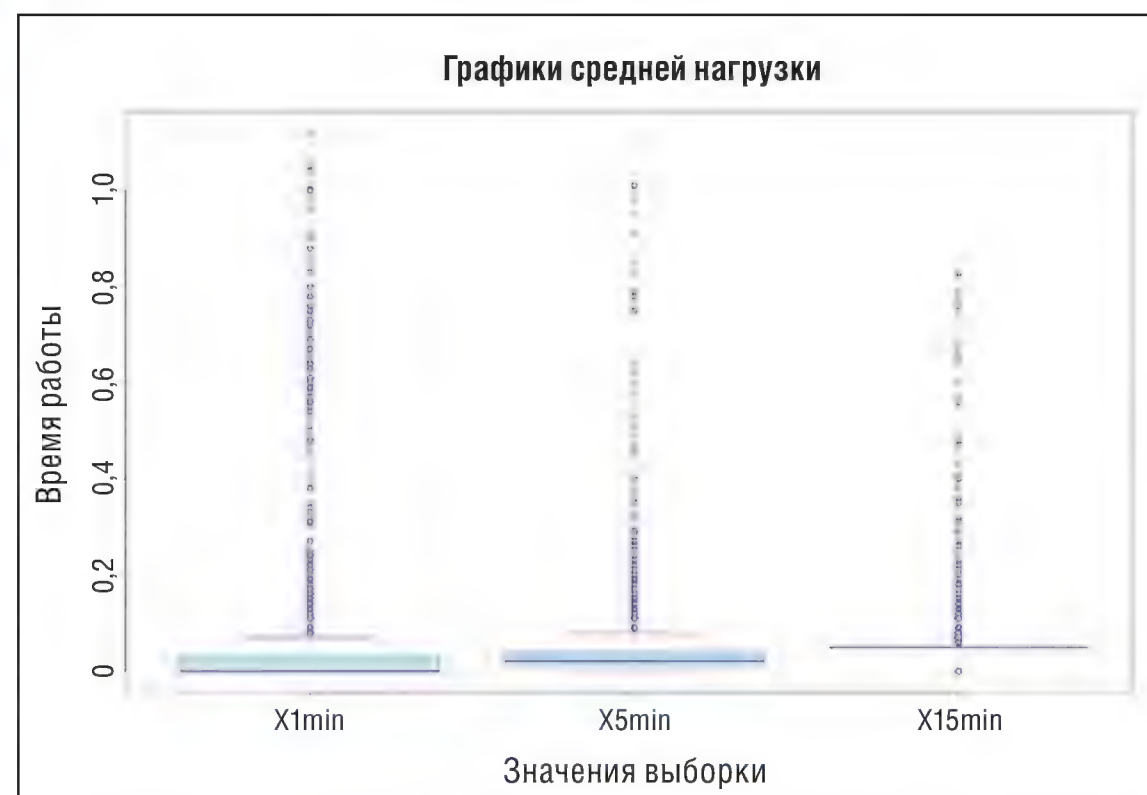
Теплокарты

Теплокарта — это способ визуализировать таблицу с числами, заменяя настоящие значения цветными клетками. Она удобна для обнаружения высот и глубин и, возможно, закономерностей. Теплокарты лучше всего подходят для небольших наборов данных. Не пытайтесь их использовать для визуализации более 500 значений: это потребует от вас более детального знания R.

Мониторинг данных для нескольких компьютеров — отличный кандидат для теплокарты. Карта на рисунке вверху слева создана по моему набору данных такими командами:

```
> data <- read.table("mapData", header=TRUE)
> data_matrix <- data.matrix(data)
> head(data)
```

	X1min	X5min	X15min	X30min
1	0.5	0.01	0.05	1.1
2	0.5	0.01	0.05	1.3
3	0.5	0.01	0.05	1.3
4	0.5	0.01	0.05	1.2
5	0.5	0.03	0.05	0.9



» **Столбчатая диаграмма** — отличный способ быстрого представления распределения, разброса и медианы набора данных.

Скорая помощь

Статистика способна ввести в заблуждение. Если человека сбила машина, за последние три часа двигавшаяся со средней скоростью 5 км/ч, то какая скорость была у машины в момент столкновения? Ответ: 100 км/ч. Никогда не путайте среднее и максимальное значения.

» **Пропустили номер?** Узнайте на с. 108, как получить его прямо сейчас.


```
6      0.5      0.01      0.05      1.3
> heatmap(data_matrix, col = heat.colors(32), Rowv=NA, Colv=NA,
margins=c(7,10))
```

Чтобы создать теплокарту в других цветах, вместо **heat.colors** используйте **cm.colors**, **topo.colors** или **terrain.colors**.

Автоматизация и сортировка в R

В моей предыдущей статье [см. Учебники **LXF191/192**, стр. 80] я объяснял, как извлечь данные мониторинга в виде текстовых файлов и обработать их вручную с помощью R. На сей раз мы автоматизируем этот процесс скриптами R.

Сначала создадим теплокарту. Каждый созданный файл изображения получит уникальное имя, чтобы сохранить хронологические данные.

Вот файл скрипта, **heatmap.R**:

```
#!/usr/bin/env Rscript
now <- format(Sys.time(), "%b%d%H%M%S")
file_base <- "heatMap"
outputfile <- paste(file_base, "-", now, ".png", sep="")
data <- read.table("mapData", header=TRUE)
png(filename=outputfile, width=1280, height=800)
data_matrix <- data.matrix(data)
heatmap(data_matrix, col = heat.colors(32), Rowv=NA,
Colv=NA, margins=c(7,10))
```

Rscript — это интерфейс для создания скриптов в R, он очень удобен для запуска кода R через *Cron*. Сделав **heatmap.R** исполняемым файлом, как вы поступили бы со скриптом *Bash* (**chmod 755**), вы сможете запросто запускать его как задание *Cron*!

Можно реализовать в R знаменитый алгоритм сортировки Bubble. Приведенный ниже код довольно медленный, но зато его легко понять, даже если вы незнакомы с алгоритмами сортировки:

```
mySort = function(set)
{
  len = length(set)
  found = 1
  while (found == 1)
  {
    found = 0
    for (k in (1:(len-1)))
    {
      if (set[k] > set[k+1])
      {
        temp = set[k]
        set[k] = set[k+1]
        set[k+1] = temp
        found = 1
      }
    }
  }
  return(set)
}
```

В этой реализации мы используем сокращение **(1:(len-1))**, которое создает все нужные циклы для сортировки Bubble:

```
> len = 10
> (1:(len-1))
[1] 1 2 3 4 5 6 7 8 9
```

Иными словами, переменная **k** берет все значения из нового набора, создаваемого при помощи быстрой ссылки **(1:(len-1))** шаг за шагом. Для проверки правильности можете создать тестовый набор данных с 100 значений от 0 до 1000:

```
> test_vec = round(runif(100, 0, 1000))
> mySort(test_vec)
```

Команда **system.time()** поможет выяснить, сколько времени идет на выполнение операции. Она похожа на команду Unix, **time**.



➤ Гистограмма хакерских атак на сайт *WordPress*, созданная автоматически из лог-файла с помощью R.

После сортировки результат **system.time()** должен выглядеть приблизительно так:

```
> set = c(1, 3, 4, 0, -1)
> mySort(set)
[1] -1 0 1 3 4
> system.time(sort(set))
user      system    elapsed
0.000      0.000      0.001
```

R на страже сервера

Обработка лог-файлов, содержащих данные web-сервера, бывает весьма непростой задачей, но R прекрасно с ней справляется! Чтобы импортировать лог-файл в R:

```
> LOGS = read.table("logfile.log", sep="«", header=F)
```

Для примера я проанализирую лог-файл с сайта на *WordPress*. Я хочу отследить запросы **POST /wp-login.php HTTP/1.1**, **POST /wp-login.php HTTP/1.0**, **GET /wp-login.php HTTP/1.1** и **GET /wp-login.php HTTP/1.0**, намекающие на попытки взлома.

Мне интересны только столбцы V4 и V6, и я могу изолировать их от переменной HACK таким образом:

```
> names(LOGS)
[1] "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10"
> HACK = subset(LOGS, V6 %in% c("POST /wp-login.php
HTTP/1.1", "POST /wp-login.php HTTP/1.0", "GET /wp-login.php
HTTP/1.0", "GET /wp-login.php HTTP/1.1"))
> names(HACK)
[1] "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10"
> HACK[1:3] <- list(NULL)
> names(HACK)
[1] "V4" "V5" "V6" "V7" "V8" "V9" "V10"
> HACK$V5 <- NULL
> HACK[3:5] <- list(NULL)
> HACK[3:4] <- list(NULL)
> names(HACK)
[1] "V" "V6"
```

Далее можно извлечь из столбца V4 день недели и создать гистограмму:

```
> newV4 <- strptime(HACK$V4, format('%d/%b/%Y:%H:%M:%S'))
> day = format(newV4, "%A")
> barplot( table(factor(day, levels=c("Пн", "Вт", "Ср", "Чт", "Пт", "Сб",
"Вс"))), xlab="День недели", ylab="Количество", col="orange",
border="lightblue", main="Попытки взлома WordPress!")
> grid()
```

Как и следует ожидать, диаграмма (вверху) указывает, что большая часть хакерских атак приходится на воскресенье, когда мониторинг системы не проводится.

И это завершает наше исследование R и его применения в анализе системных данных. Всегда помните, что теплокарта или гистограмма — всего лишь картинка: смысл любому созданному вами графику придают ваши данные. **LXF**

ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru, и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

- 1 Сортировку фотографий
- 2 Надежное удаление данных с флэшек
- 3 Работу с «чужеродными» дисками
- 4 Безопасность и шифрование облака
- 5 Решение проблем с btrfs
- 6 Копирование MP3 по порядку

1 Сортировка фотографий

В У меня более 40 каталогов, содержащих несколько тысяч фотографий, которые я хотел бы отсортировать, причем отсортировать по дате, и желательно по дате создания. Но, насколько я понимаю, в Linux на данный момент нет даты создания, есть только дата изменения.

Можно ли как-то скопировать их в один большой каталог и запустить команду сортировки, чтобы после нее фотографии отображались в окне в правильном порядке, и я мог бы скопировать их в отдельные каталоги по времени создания? Я попробовал поискать в Интернете, но ничего очевидного не нашел. Буду очень благодарен за любую помощь.

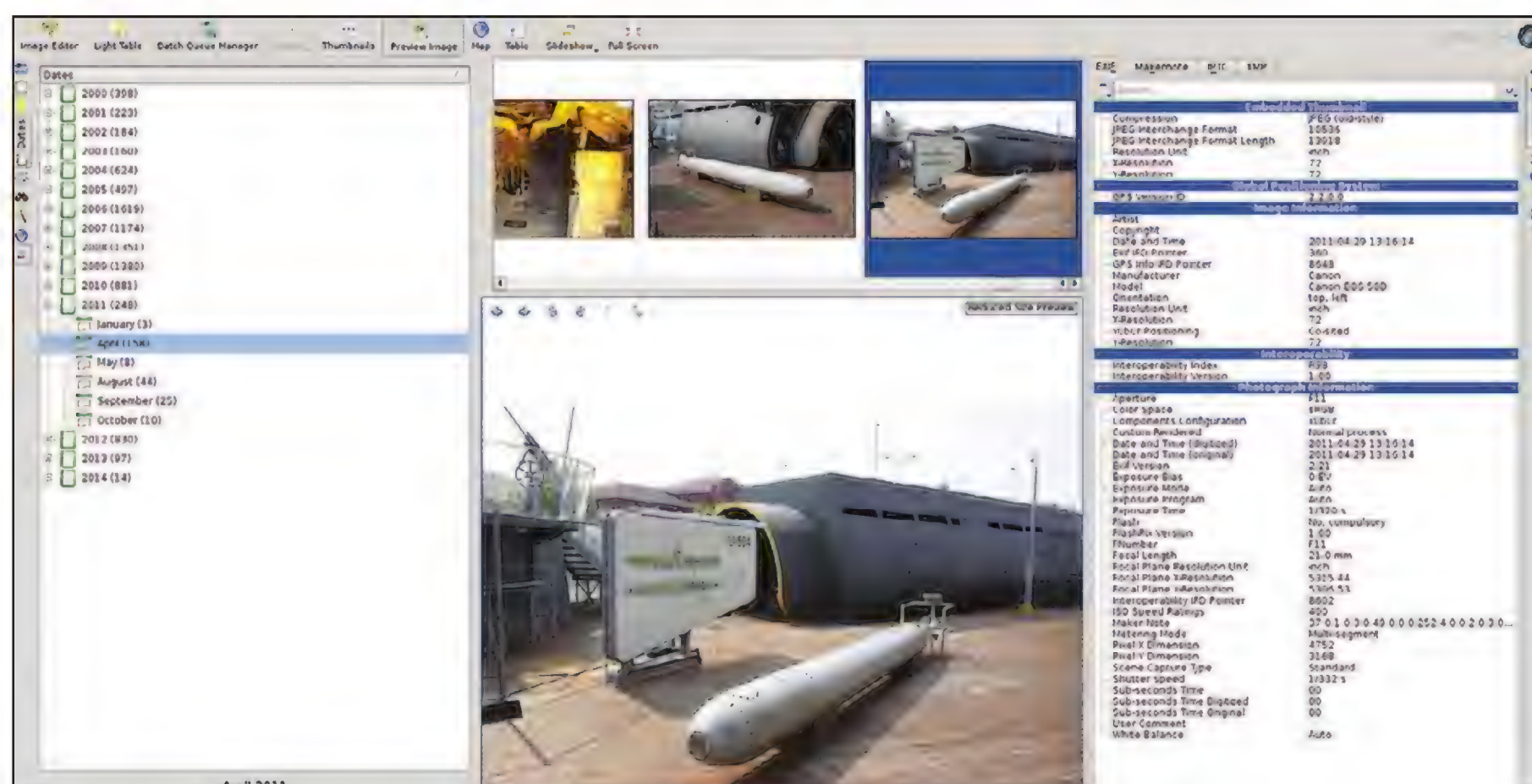
GeordieJedi, с форумов

Наряду с самим изображением в файле с цифровой фотокамеры содержатся и другие данные, называемые EXIF (подробности см. на <http://bit.ly/EXIFDefined>). Они могут включать данные о выдержке, фокусное расстояние объектива, значение ISO, использование вспышки и даже координаты GPS. Самое главное — данные EXIF содержат время создания снимка. С этой информацией могут работать многие программы для работы с цифровыми изображениями.

В *DigiKam*, которую предпочитаю я, эта информация отображается с каждой миниатюрой, а при выборе изображения показывается дополнительная информация. В ней также можно просматривать фотографии по дате съемки в виртуальных каталогах и подкаталогах с годами и месяцами независимо от того, как фотографии фактически расположены на диске.

Если Вам нужно разобрать таким образом несколько тысяч файлов, то придется писать скрипт, и программа, которая Вам нужна — *ExifTool*. Эта программа считывает (и записывает, но в данном случае это неважно) тэги EXIF. Команда

```
exiftool photo.jpg
```



Благодаря тэгам EXIF *digiKam* может отображать все фотографии на диске в виртуальных каталогах, разбитых по дате съемки.

выведет все тэги EXIF для указанного файла, но нам интересен только один, `DateTimeOriginal`, прочитав который можно командой

```
exiftool -DateTimeOriginal photo.jpg
```

Вывод *ExifTool* можно изменить в соответствии со своими потребностями. Загляните на страницу, чтобы узнать подробности работы параметров `-p` и `-d`, а требуемую Вам информацию поможет получить следующая команда:

```
exiftool -d '%Y %m' -p  
'$FileName$DateTimeOriginal' *.jpg
```

Для каждой фотографии она в одной строке выводит три параметра: имя файла, год и месяц съемки. Теперь можно воспользоваться коротким скриптом для перемещения всех файлов:

```
exiftool -d '%Y %m' -p '$FileName$DateTimeOriginal'  
*.jpg | while read NAME YEAR MONTH; do  
  mkdir -p $YEAR/$MONTH  
  mv -i $NAME $YEAR/$MONTH  
done
```

Параметр `-i` у команды `mv` означает, что команда запросит у Вас подтверждение, если файл с таким именем уже существует. Это предохранит Вас от досадной опечатки, когда команда попытается скопировать все файлы в один. Вместо маски можно указать каталог для обработки всех подходящих файлов в этом каталоге или добавить `-r`, чтобы команда спустилась и в подкаталоги.

Для обработки всей коллекции подойдет примерно такая команда:

```
exiftool -d '%Y %m' -p  
'$FileName$DateTimeOriginal' -r ~/Photos
```

Позаботьтесь о том, чтобы файлы перемещались в какой-либо другой каталог; в противном случае Вы будете пытаться переименовать файлы сами в себя.

2 USB из ничего

Несколько недель назад команда немецких исследователей продемонстрировала, что USB-флэшки небезопасны, даже после переформатирования! Подозреваю, что то же относится и к SD-картам. Возможным решением было бы обнуление флэшки командой

```
sudo cat /dev/zero >/dev/sdb
```

где `sdb` — флэшка. Эффективно ли это решение для удаления любого вредоносного ПО?

Проблема в том, что после «обнуления» флэшка не читается в Linux. (В Windows ее все еще видно, и она предлагает переформатировать ее — почему же в Linux это невозможно?) А если возможно, то как переформатировать «обнуленную» флэшку? И как смонтировать флэшку после переформатирования?

Оливер [Oliver]

После переформатирования не безопасно ни одно устройство, поскольку при переформатировании с устройства удаляется очень немного данных, если вообще удаляется. Заполнение устройства нулями — разумный способ перезаписи всех предыдущих данных. Но, записав в `/dev/sdb` вместо `/dev/sdb1`, Вы обнулили все устройство, а не только файловую систему. Это означает, что Вы заодно стерли таблицу разделов диска. Отформатировать USB-диск без разбиения на разделы возможно, хотя это скорее свойственно Windows. Разбивать флэшку на разделы нет смысла, так как ее можно просто отформатировать целиком, командой

```
sudo mkfs.vfat -l -F 32 /dev/sdb
```

Параметр `-l` необходим, потому что утилита *mkfs* обычно не создает файловую систему

на устройстве, не разбитом на разделы (за исключением дискет, если Вы помните, что это такое). Запуск **mkfs** для всего диска обычно является ошибкой, которая способна стереть все данные на диске, и по умолчанию это запрещено; с параметром **-l** данный запрет снимается. Если Вы хотите разбить устройство на разделы, сначала создайте один раздел на всем диске командой **fdisk** и затем запустите указанную выше команду без параметра **-l** и с **sdb1** вместо **sdb**.

Однако разбить устройство на разделы и отформатировать проще с помощью *GParted* — потребуется всего один шаг. Если в Вашем дистрибутиве эта программа по умолчанию не установлена, то найдется в его программных библиотеках. Обратите внимание, что речь идет о *GParted*, а не о *parted*. Последняя — утилита командной строки, которая еще менее дружелюбна к новичкам, чем **fdisk**.

В большинстве настольных дистрибутивов есть автомонтировщики для съемных устройств, поэтому после форматирования флэшки выньте ее и вставьте обратно, и она будет распознана и смонтирована. Если Вы хотите смонтировать ее из командной строки, воспользуйтесь командой

```
mount /dev/sdb (or sdb1) /media/usbstick
```

где **/media/usbstick** — каталог, в который должна смонтироваться флэшка; он должен уже существовать.

3 Старожил Atari

У меня есть компьютер Atari ST из середины 1980-х с блоком внешнего жесткого диска, состоящего из жесткого диска SCSI с 50-контактным разъемом и «адаптера хоста», который связывает стандартный интерфейс SCSI с экзотической Atari-версией SCSI (ACSI).

Я хочу проверить, работает ли диск, и если да, сделать его образ на другом диске. Для этого у меня есть интерфейс Adaptec PC PCI SCSI.

Моя идея состоит в том, чтобы сделать образ SCSI-диска в Linux как файл на жестком диске или флэшке. Емкость диска Atari — около 20 МБ, поэтому я смогу сохранить бесчисленное количество резервных копий даже на самых скромных внешних устройствах.

Единственное, что меня беспокоит — возможное повреждение диска Atari, если мой Linux попытается смонтировать его автоматически. Меня беспокоит, что это может произойти даже с Live CD, которые загружают рабочий стол с браузером и т.д. Не беспокоюсь ли я попусту? Или нужно использовать дистрибутив, который загружается прямо в терминал?

На этом диске хранится незаменимый материал, скопившийся более чем за 30 лет, и, думаю, Вы понимаете мое беспокойство. Может ли файловая система диска (вероятно, Atari ST GEMDOS) поддерживаться в Linux?

SiriusHardware, с форумов

Автомонтировщики — довольно «нерешительные» программы. Они пытаются смонтировать файловую систему, и если не получается, сдаются. Если файловая система не смонтирована, то запись в нее невозможна, а значит, Вы в безопасности. Автомонтировщики также не ограничены настольными дистрибутивами — они могут запускаться и при загрузке в терминал. Я бы посоветовал воспользоваться System-RescueCd. При загрузке он пытается смонтировать все подключенные устройства, чтобы найти файл **sysrcd.dat**, но оставляет свои попытки, если на разделе нет «стандартной» файловой системы. Для дополнительной безопасности можно воспользоваться опцией загрузки **skipmount** для пропуска одного или нескольких разделов. Сначала зайдите в меню BIOS, чтобы проверить порядок дисков, затем нажмите Tab в меню загрузки и добавьте, например, следующее:

```
skipmount=/dev/sdb1 skipmount=/dev/sdb2
```

Терминалы и суперпользователи

Мы часто предлагаем в качестве решения проблемы ввести те или иные команды в терминале. Хотя обычно то же самое можно сделать с помощью графических утилит дистрибутива, такие решения будут слишком конкретными (будут зависеть от дистрибутива). Команды в терминале более гибкие и — самое главное — ими можно пользоваться во всех дистрибутивах. Команды настройки системы часто нужно выполнять от имени суперпользователя, называемого также **root**. Существует два основных способа это делать, в зависимости от используемого дистрибутива. Во многих дистрибутивах, особенно в Ubuntu и его производных, перед командой можно написать **sudo** — при этом будет запрошен пароль пользователя, и ему будут предоставлены привилегии **root** только на время выполнения команды. В других дистрибутивах применяется команда **su**, для использования которой требуется ввести пароль **root** и которая предоставляет полный доступ **root** до того момента, пока вы не наберете **logout**. Если в вашем дистрибутиве используется **su**, запустите ее один раз и выполняйте любые заданные команды без предшествующей **sudo**.

Обратите внимание, что здесь Вы указали отдельные разделы, а не весь диск. После загрузки в SystemRescueCd можете скопировать данные командой **dd**. Так как **dd** считывает «сырые» данные с диска, то для нее не важны разделы или файловые системы.

У ядра Linux есть параметры монтирования для таблиц разделов и файловых систем Atari, поэтому Вы должны успешно смонтировать диск и скопировать нужные данные — хотя я посоветовал бы сначала скопировать диск командой **dd**. Однако у Вас могут возникнуть сложности »

Коротко про...

Паника ядра

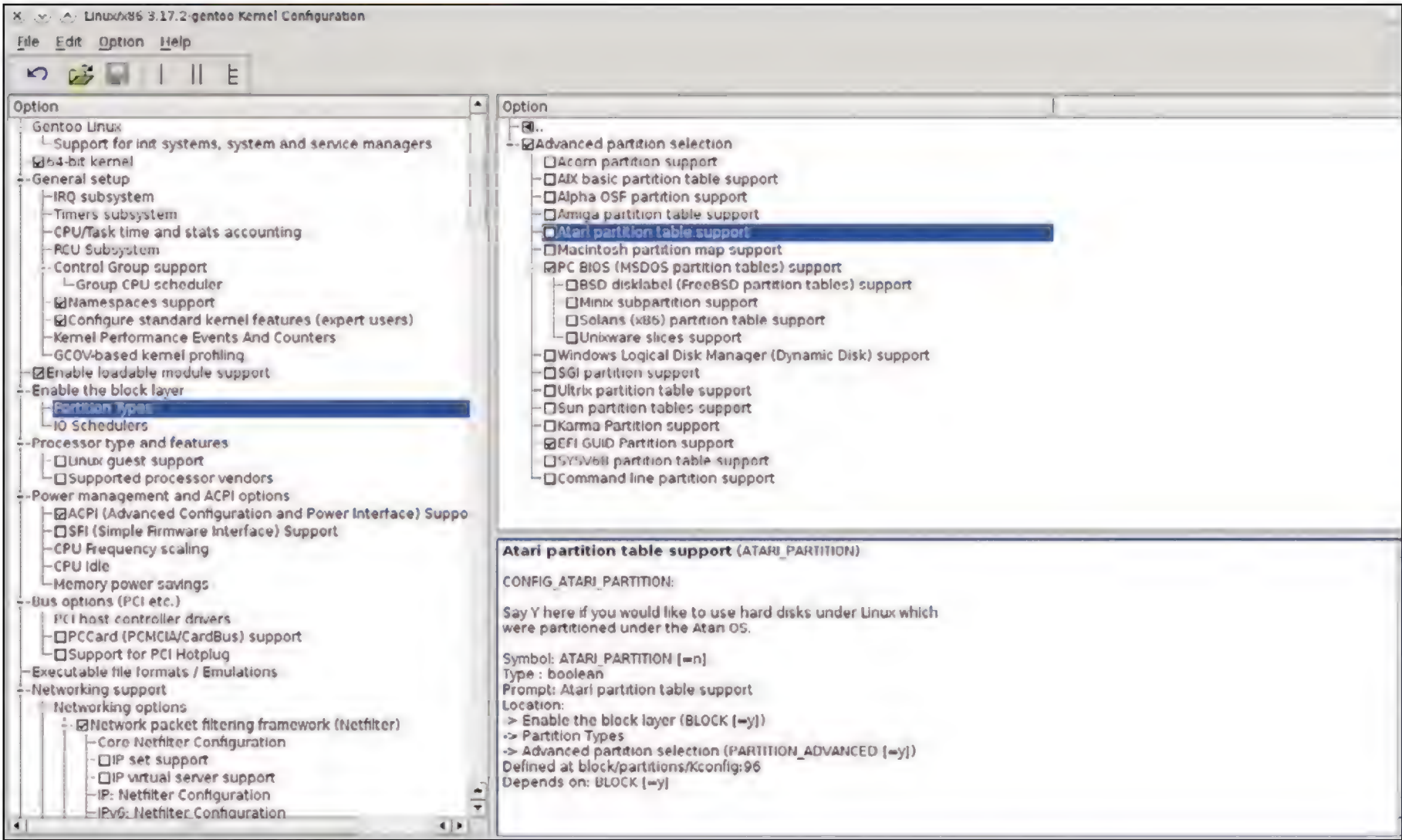
Иногда ядро Linux сталкивается с ситуацией, с которой не может справиться, и вопреки совету в названии величайшей из когда-либо написанных книг паникует. Это проявляется в выводе непонятных сообщений в консоль (иногда они скрыты за симпатичным экраном загрузки, и тогда вам реально не повезло) и мигании светодиода Caps Lock. Если ядро успело смонтировать корневую файловую систему в режиме чтения/записи, оно также выводит на диск диагностическую информацию, «дамп ядра». Теоретически с ядром из вашего дистрибутива вы никогда не должны увидеть это сообщение. Если вы видите его

с ядром, собранным вручную, это обычно означает, что вы забыли встроить поддержку контроллера жесткого диска и файловой системы, используемой корневой файловой системой, и поэтому ее не удастся смонтировать.

Тут вам простительно подумать, что всё пропало; но ядро всё еще может совершить последний героический поступок, даже если вы его убьете. Ядро можно попросить перезагрузить систему после возникновения паники. При добавлении параметра **kernel.panic=10** в **/etc/sysctl.conf** система будет перезагружена через 10 секунд после начала паники, но для этого необходимо, чтобы система смонтировала

корневую файловую систему и можно было прочесть этот файл. Более надежный вариант — добавить **panic=10** к параметрам ядра в меню *Grub*.

Вы можете поинтересоваться: а стоит ли инициировать перезагрузку, если после нее снова возникнет паника, и система снова перезагрузится, и так до бесконечности? Ответ — в возможности *Grub* воспользоваться последней рабочей конфигурацией ядра. С этой настройкой паника ядра вызовет перезагрузку в последнюю конфигурацию ядра, о которой точно известно, что она работает. Это настоящий дар небес при загрузке новой версии ядра на удаленном компьютере — он создает гораздо меньше неудобств, чем если вам приходится звонить и просить кого-нибудь перезагрузить систему, потому что вы испортили ядро.



Linux умеет читать таблицы разделов для многих старых (и новых) систем — даже если вам не повезло, и у вас Atari вместо Amiga!

с поиском Live CD с ядром, у которого была бы опция ATARI_PARTITION (у ядра SystemRescue-CD ее нет). Между прочим, 50-контактный разъем был стандартным для SCSI. Думаю, у меня еще есть несколько SCSI-кабелей — я их не выбросил «про запас на всякий случай».

4 Чугунное облако

В Сейчас я пользуюсь Dropbox в качестве межплатформенного облачного хранилища, но если верить Сноудену, это не слишком хорошо. Он советует SpiderOak.

После некоторого исследования я нашел независимые от платформы (Microsoft, Apple, Linux, iOS, Android), защищенные облачные хранилища: SpiderOak (2 ГБ, бесплатно), Wuala (5 ГБ, \$0,99 в месяц), Mega (50 ГБ, бесплатно) и Tresorit (5 ГБ, бесплатно). Какое бы посоветовали вы и почему?

Или можно пока остаться на Dropbox, но начать шифровать свои файлы. Какую программу для шифрования вы посоветуете: TrueCrypt или 7-Zip? Она не должна зависеть от платформы. А также какой менеджер паролей (не зависящий от платформы) вы бы посоветовали?

JohanM

В Вашем списке отсутствует один заметный вариант: ownCloud. Если Вам нужна безопасность, запустите ownCloud локально или установите туда VPS. Руководство по настройке ownCloud недавно публиковалось [см. «Учебники» LXF190, стр. 64].

В любой системе с шифруемыми хранилищами кто-то должен знать ключи шифрования. SpiderOak более защищен, чем Dropbox, так как шифрование выполняется локально ключами, хранимыми только в Вашей системе. Другие системы могут позволять обращаться к Вашим данным кому-то с нужным уровнем привилегий — будь то недобросовестный сотрудник, который кому-то их продает, или некто другой, действующий по указке правительственного агентства.

Недостаток этого подхода в том, что обращаться к данным можно только с устройства, где есть ключи. Защита данных — это не только сохранность тайны Ваших данных, но и их безопасность. OwnCloud дает Вам полный контроль над данными, но также принимает всю ответственность за безопасность системы от аппаратных или программных ошибок. Большие хранилища, вроде упомянутых Вами, обладают резервным копированием и избыточностью, но Вы получаете то, за что платили. Что касается TrueCrypt, он больше официально не поддерживается. Для шифрования отдельных файлов или архивов воспользуйтесь GPG или PGP, так как они доступны на всех платформах. Для управления паролями я пользуюсь KeePass (<http://keepass.info>). У этого приложения есть клиенты для настольных систем и мобильных платформ, а все данные хранятся в одном зашифрованном файле, который легко хранить на облаке, например, в Dropbox.

5 Неработающая btrfs

В Я некоторое время пользуюсь btrfs, но недавно она начала работать с ошибками. Вскоре после загрузки все подтома btrfs переходят в режим только для чтения. После перезагрузки какое-то время все нормально, затем проблема повторяется. В сообщениях в /var/log/ я ничего не нашел, но dmesg возвращает следующие сообщения:

```
[ 291.245123] parent transid verify failed on 11234344960 wanted 247156 found 250750
[ 291.246245] parent transid verify failed on 159174590464 wanted 247379 found 250606
[ 291.246814] parent transid verify failed on 159174590464 wanted 247379 found 250606
```

Вслед за ними идет трассировка стека с упоминанием btrfs. Запуск fsck не помог решить проблему. Нейл Дэвидсон [Neal Davidson]

Btrfs — хорошая файловая система, и в будущем вполне может появиться во многих дистрибутивах по умолчанию,

Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, поскольку проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам необходимо знать как можно больше.

Если у вас появляется сообщение об ошибке, приведите его точный текст и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все установленные устройства.

Если Linux уже запущен, можете применить для этого отличную программу Hardinfo (<http://sourceforge.net/hardinfo.berlios>) — она сохранила подробную информацию об устройствах и о состоянии системы в HTML-файле, который вы сможете приложить к своему письму.

Не уступающий в удобстве альтернативный вариант — lshw (<http://ezix.org/project/wiki/HardwareLiSter>). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если вы не хотите или не можете их установить, выполните следующие команды в терминале от имени root и приложите файл system.txt к письму. Это здорово поможет диагностике.

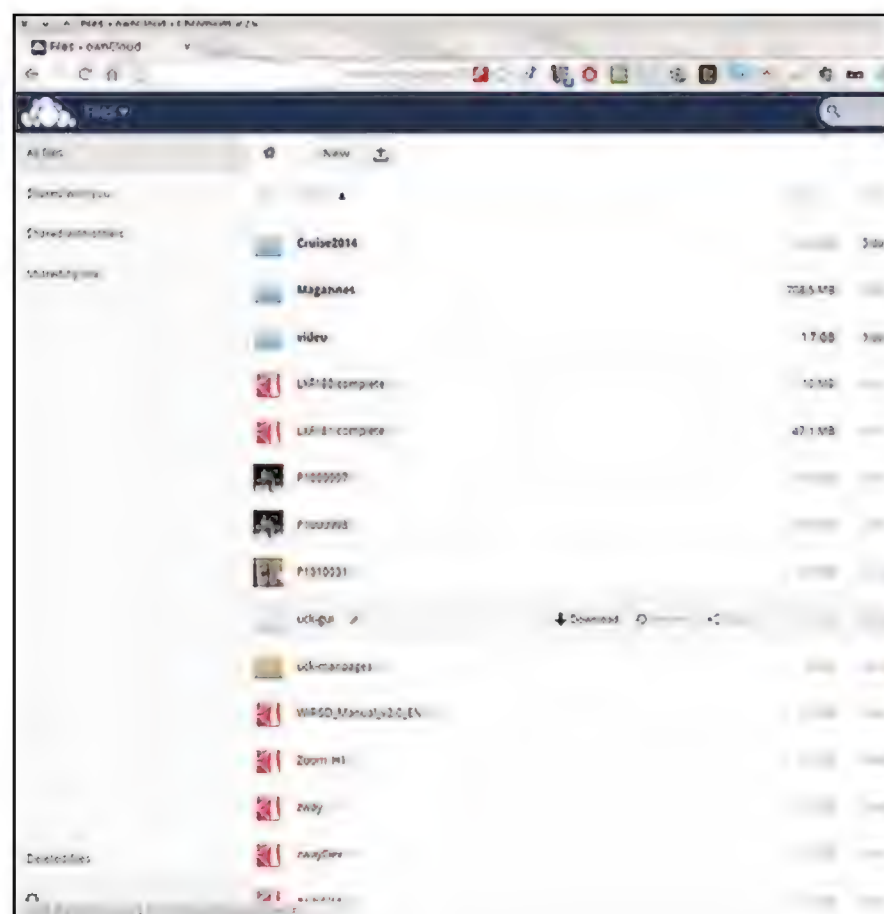
```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

но она все еще находится в активной разработке, хотя и считается стабильной. Это означает, что следует всегда использовать последнюю версию ядра для своего дистрибутива и регулярно делать резервное копирование. Ошибки, которые Вы видите, вызваны несоответствием кэша, используемого btrfs, и фактической файловой системой, возникшим, возможно, из-за отключения питания и аппаратной ошибки. Как и с некоторыми другими файловыми системами, когда btrfs определяет ошибку, она старается защититься от дальнейшего повреждения, перемонтируя себя в режиме только для чтения.

Первое, что нужно сделать — перемонтировать файловую систему с параметрами recovery,nospace_cache,clear_cache

Если это корневая файловая система, добавьте параметры к fstab и перезагрузитесь. Параметр recovery означает, что btrfs надо попробовать исправить ошибки; остальные параметры велят игнорировать кэш. Восстановление может занять некоторое время. Это единственный вариант, не вызывающий повреждения данных, и если он не сработает, следующим шагом должно стать резервное копирование данных командой btrfs restore DEVICE-NODE BACKUPDIRECTORY

с соответствующим устройством и местом назначения, которое должно находиться на другом диске с файловой системой Linux. Если хотите также сделать копии снимков файловой системы, добавьте параметр -s. Используйте Live CD, так как эти действия нужно выполнять с размонтированной файловой системой, и последнюю



➤ Если вы не доверяете платным и бесплатным хранилищам, всегда можно запустить свое собственное с помощью *ownCloud*.

версию ядра. Я бы воспользовался SystemRescueCd с альтернативным ядром. Теперь, все еще с Live CD, попробуйте очистить лог файловой системы командой:

```
btrfs-zero-log DEVICE-NODE
```

Перезагрузитесь и посмотрите, помогло ли это решить проблему. Если нет, последний вариант — проверить диск с Live CD командой **fsck** — не *fsck.btrfs*, а основной утилитой *btrfs*:

```
btrfs check --repair DEVICE-NODE
```

Эта команда потребует немало времени, но она проверит и восстановит Ваш диск. В случаях серьезного повреждения можете добавить один из параметров **--init-extent-tree** или **--init-csum-tree**, но если файловая система повреждена, она скорее всего вообще не смонтируется. В ходе операций с **btrfs-zero-log** и **--repair** Вы можете потерять некоторые данные — некоторые файлы могут вернуться к предыдущим версиям — но, надеюсь, потери будут минимальными.

6 MP3 по алфавиту

В У меня есть немного музыки в каталоге с подкаталогами по исполнителям и альбомам. Когда я копирую эту музыку на флэшку для автомобиля, то хочу, чтобы она копировалась в том же порядке, который я вижу на компьютере, т.е. в алфавитном. Но MP3-файлы, кажется, копируются в том порядке, в котором файлы расположены на диске? Я хочу, чтобы музыка воспроизводилась в том порядке, в котором мне надо, а не в каком-то «случайном».

nuxguu

○ Это частая проблема некоторых простых MP3-плееров. Файлы сохраняются на диск или на флэшку без особенного порядка (как говорят на голосованиях по телевизору), и программе приходится самой делать список содержимого каталога для сортировки.

Таким способом использовать диск будет быстрее и эффективнее.

Вам могло бы помочь копирование в алфавитном порядке, но это все равно не гарантирует, что файлы добавятся в список FAT диска в требуемой Вам последовательности. Даже если повезет — ну, а если Вы потом захотите добавить новые файлы? Пришлось бы переформатировать диск и копировать все снова, даже для добавления одной песни или альбома.

Эта проблема настолько частая, что есть программа для ее решения — *FATSort* (<http://fatsort.sourceforge.net>). Запустите ее после копирования файлов на флэшку, и она пересортирует список FAT, расположив файлы в алфавитном порядке:

```
fatsort -c /dev/sdb1
```

Учтите: во время запуска *FATSort* флэшка не должна быть смонтирована, в противном случае это приведет к повреждению файловой системы. Параметр **-c** включает сортировку без учета регистра, и именно это, наверное, Вам и нужно. На *man*-странице *FATSort* описаны несколько параметров сортировки. Например, удобен параметр **-l**, так как он отображает текущий порядок файлов на устройстве. Порядок файлов на смонтированном устройстве также можно просмотреть старой доброй **ls**. По умолчанию **ls** сортирует вывод в алфавитном порядке, но это можно отключить и увидеть естественный порядок файлов на диске, запустив **ls -U**. **LXF**



Часто задаваемые вопросы

Печать

➤ Что такое CUPS?

CUPS (Common Unix Printing System — Общая система печати Unix) — набор драйверов и утилит для полной поддержки, управления и работы с принтерами в Linux и других операционных системах на основе UNIX.

➤ То есть это драйвер принтера?

Ну, да, но и гораздо больше. CUPS предоставляет «переносимый уровень печати» между приложениями и принтером. Обычно он включает драйверы, но также предоставляет все необходимое для того, чтобы программы могли печатать.

➤ Значит, мне придется маяться с командной строкой и редактировать файлы настройки, чтобы заставить принтер работать?

Вовсе нет. В CUPS есть собственные графические утилиты конфигурации, которые работают через

web-браузер. Откройте в своем любимом браузере <http://localhost:631> (может потребоваться ввод своего пароля или пароля root), и вы окажетесь на домашней странице CUPS. На ней можно просматривать список принтеров, добавлять и удалять принтеры, а также работать с очередями печати (ожидаящими выполнения заданиями на печать). Здесь же есть документация.

➤ Почему браузер — вместо более обычных графических программ?

Использование web-интерфейса означает, что вам не придется ставить никаких дополнительных графических пакетов: вам даже не нужен X-сервер. С интерфейсом можно работать и в текстовом браузере вроде *Elinks* или в браузере, запущенном на другом компьютере.

➤ А ведь это немного небезопасно?

Возможно, но с настройками CUPS по умолчанию подключения разрешаются только с **localhost**. Эту настройку можно изменить, разрешив подключения с локальной сети

(открывать доступ через Интернет в общем случае не рекомендуется). Также можно управлять тем, какие пользователи имеют доступ к каким частям конфигурации — пользователям можно разрешить работать со своими заданиями на печать и запретить удалять свои принтеры.

➤ Какую роль здесь играет Gimp-print и что общего у GIMP с печатью?

Gimp-print — набор драйверов, изначально предназначенный для использования с *GIMP*. Хотя *GIMP* хорошо работает с CUPS, некоторые принтеры могут дать лучший результат с драйверами Gimp-print.

Эти драйверы сейчас также работают и с CUPS, поэтому Gimp-print можно использовать в качестве расширенного набора драйверов, которые работают со всеми программами, использующими для печати CUPS, а не только с *GIMP*. Чтобы избежать этой двусмысленности, проект сейчас называется Gutenprint, но в некоторых дистрибутивах всё еще есть пакеты Gimp-print.

➤ А что такое Gutenprint?

Это новое название Gimp-print. В зависимости от того, насколько «современный» дистрибутив вы используете, у вас будет Gimp-print 4.x или Gutenprint 5.x. Поскольку у Gimp-print гораздо больше общего с печатью, чем с *GIMP*, новое имя не так вводит в заблуждение, хотя любое изменение имени способно вызывать кратковременную неразбериху, и это само по себе сбивает с толку.

➤ Как узнать, поддерживается ли мой принтер?

Вашим первым пунктом назначения должен стать сайт www.linuxprinting.org, где содержится исчерпывающая база данных с информацией о том, насколько хорошо (или плохо) поддерживается каждый принтер, и с советами по применяемым драйверам. Прежде чем покупать принтер, лучше навести подробные справки, не то может оказаться, что ваше новое приобретение пригодно только в качестве пресс-папье.



LXF Hot Picks



Александр Толстой

CherryTree » Lunar Linux » OpenSSH » Snapper » Remmina » Wal Commander
GitHub Edition » rTorrent » Neverball » O A.D. » Konversation » Tomahawk

Сигает с узла на узел, как кибер-архар, обегая высоты Пика Сети в поисках самых сочных программ с открытым кодом.

Инструмент для заметок

CherryTree

Версия: 0.35.3 Сайт: www.giuspen.com/cherrytree

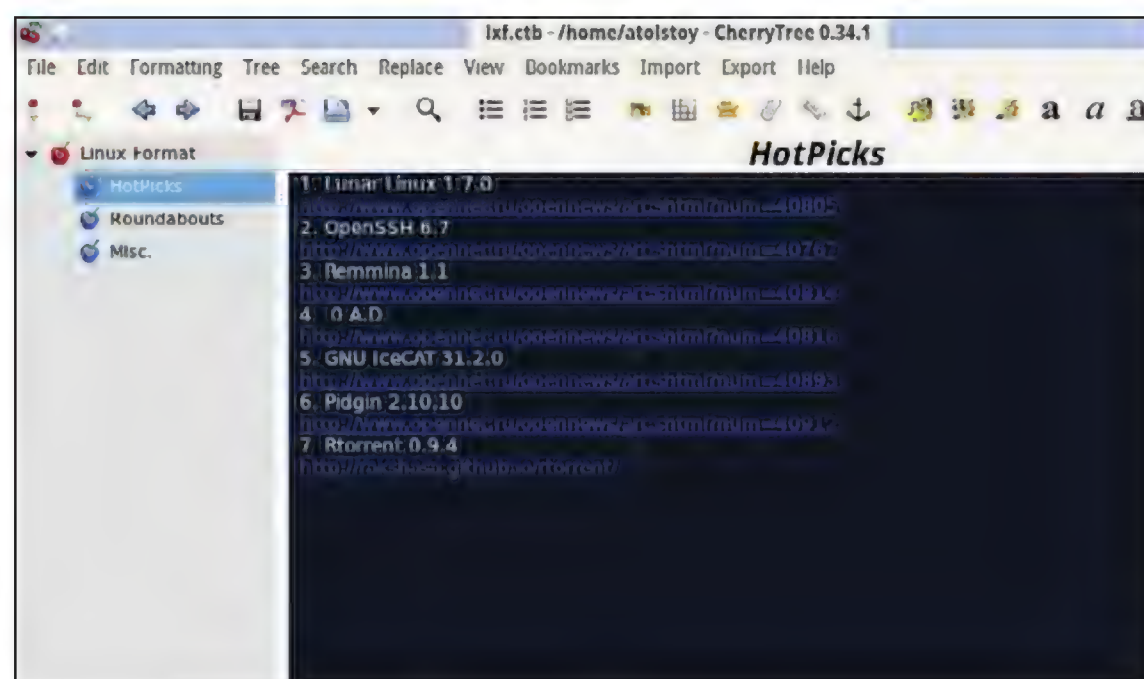
Если вам нравится сохранять заметки в виде простых текстовых файлов, знайте, что в некий момент вся эта мешанина заставит вас или завести спецприложение для заметок, или упорядочить свои файлы в некое подобие структуры, именую файлы и размещая подпапки по какой-либо закономерности. Однако есть и альтернатива: *CherryTree*.

Текстовый редактор *GTK*, *CherryTree* требует, чтобы вы сначала определили в своем дереве данных узлы и подузлы; и только потом вы заполняете эти узлы контентом. Вы можете рассматривать документ *CherryTree* как корень дерева;

каждый его узел — ветвь этого дерева, а подузлы — мелкие веточки.

CherryTree поддерживает форматирование текста, выделение синтаксиса, ссылки, изображения и даже несложные таблицы. Конечно, он не столь мощный, как полнофункциональный текстовый редактор, но такого и не требуется: это ваш внутренний инструмент для создания сложных структур данных в первую

«Ваш инструмент
для создания сложных
структур данных.»



» *CherryTree* похожа на программу создания web-страниц, но создает ощущение работы с базы данных. Здесь много хорошего.

очередь для использования на вашем рабочем столе.

Вы можете хранить свои данные как файл XML или как базу данных *SQLite*. Диалоговое окно сохранения даже позволяет вам в любом из файловых форматов защитить свой файл паролем (файл будет зашифрован).

Есть множество областей, в которых *CherryTree* отлично проявляет себя, но самые примечательные — импорт и поиск. Например, у вас уже есть текстовые файлы, которые вы хотите привести в порядок, или заметки в других приложениях, скажем, в *Tomboy* или *KeepNote*. *CherryTree* поможет вам конвертировать их в более похожий на базу данных формат — просто откройте меню *Import* и посчитайте, сколько поддерживается источников!

Перенеся данные, можете заняться расширенными опциями поиска *CherryTree*. Как только все будет сохранено в виде одного файла XML или *SQLite*, вы сможете легко найти любое слово или последовательность букв независимо от того, где они находятся: поддерживаются даже регулярные выражения.

Чтобы установить *CherryTree*, просто загляните в стандартный репозиторий своего дистрибутива; это приложение достаточно известно и имеется в пакетах для многих дистрибутивов Linux.

Решившись скомпилировать его из исходника, будьте готовы предоставить ему пакеты разработки Python, поскольку *CherryTree* использует некоторые модули Python и его привязки к *GTK2*.

Исследуем интерфейс CherryTree

Содержание

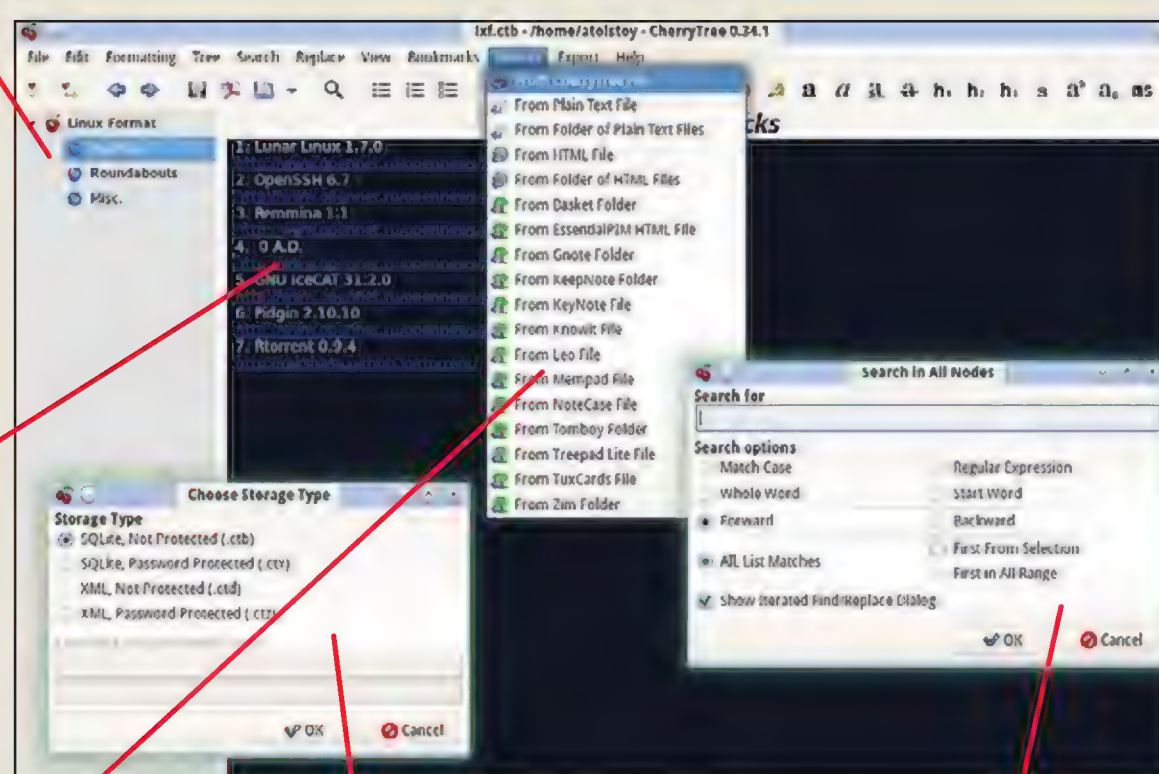
Можете использовать эту левую панель для навигации по узлам и подузлам. Вы также можете здесь создавать, модифицировать или удалять узлы, либо делать это через контекстное меню или с помощью горячих клавиш.

Основная область

Именно сюда вы вводите и здесь форматируете контент. Две цветовых схемы по умолчанию — это светлый фон с темным текстом и наоборот. Проверка орфографии осуществляется с помощью *python-enchant*.

Фильтры импорта

Почти все основные приложения и сервисы для заметок доступны в качестве опции импорта — по крайней мере те, в который открытой код.



Типы сохранения

Вы можете выбрать файловый формат XML или *SQLite*. И тот, и другой можно защитить паролем.

Расширенный поиск

Поиск — сильная сторона *CherryTree*. Результатами можно управлять в отдельном всплывающем окне.

Дистрибутив Linux

Lunar Linux

Версия: 1.7.0 Сайт: www.lunar-linux.org

Порой нелегко остановить выбор на какой-то одной из жемчужин в лапце дистрибутивов Linux. Но если вам нужен надежный, быстро развивающийся дистрибутив с самыми современными программами и отличным менеджером пакетов, Lunar Linux вполне подойдет.

Начинаете вы с загрузки ISO размером менее 400 МБ, который предлагает текстовую программу установки для базовой системы. Процесс установки довольно долгий — 14 шагов под надзором мастера — но он прост, и включает разбиение диска на разделы; настройку точек монтирования; установку загрузчика; создание пользователей и все прочее, необходимое для создания загружаемого Lunar Linux, а тот затем предложит вам инструменты для компиляции и установки приложений, которые вам (обязательно) понадобятся.

Загрузившись в новенький дистрибутив, самое время войти в Moonbase — иерархический каталог программ. Он похож

на Ports во FreeBSD или Portage в Gentoo. Для установки программ Lunar Linux использует собственный уникальный менеджер пакетов *Lin*. В пользовательской системе он сидит в `/var/lib/lunar/moonbase`. Программы разделены на категории, и каждая именуется «модулем». Например, для установки среды рабочего стола вы просто вводите `lin kde4`, или `lin gnome3`. Удаление пакетов ничуть не сложнее: их уберет `lrm kde4`.

Менеджер пакетов *Lin* был создан ради устранения «ада зависимостей», и достигает этой цели через достаточно строгие требования к модулям в Moonbase. Как только новый модуль утверждается и тестируется, он становится доступен для пользователей. Для создания

«Надежный дистрибутив с отличным менеджером пакетов.»



➤ Если вы устали от брендинга и вариаций, предлагаемых большинством производителей Linux, попробуйте базовый Lunar Linux.

модуля, который вы выбрали, *Lin* использует скрипты и метаданные модулей, автоматически скачивая соответствующие исходники и зависимости. Процесс компиляции может занять немало времени, особенно для таких больших проектов, как KDE или *Chromium*, но вы получаете идеально надежную и гибкую систему, соответствующую вашим потребностям. Создание собственных модулей — неплохой способ изучить принципы управления пакетами и, вероятно, более достижимый, чем с помощью более сложных технологий вроде Emerge в Gentoo.

Безопасная оболочка

OpenSSH

Версия: 6.7 Сайт: www.openssh.com

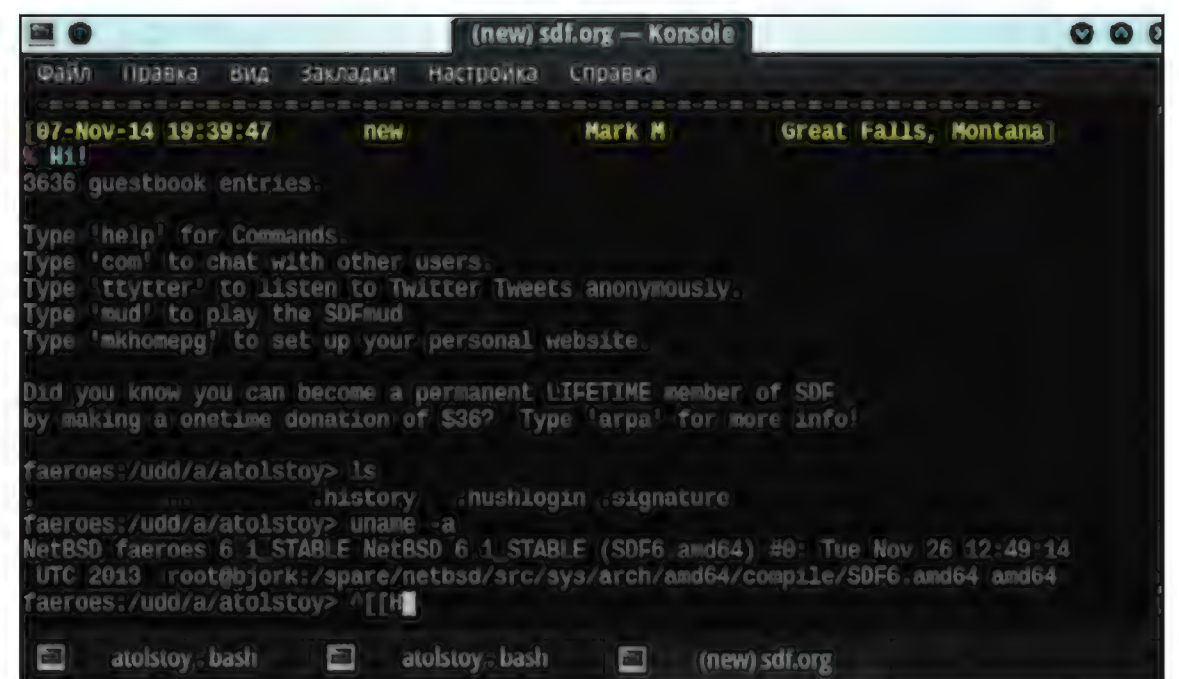
OpenSSH (OpenBSD Secure Shell) всегда была той программой, с которой редко сталкиваются домашние пользователи. Однако многие из нас в той или иной степени сисадмины, даже если нам приходится только заходить в NAS или на внешний диск, присоединенный к домашнему роутеру, или удаленно посещать компьютер своей бабушки. Для этого применяются профессиональные инструменты администрирования, и *OpenSSH* — один из них.

OpenSSH обеспечит вам безопасный и зашифрованный доступ к удаленным системам — или через Secure Shell (SSH), или Secure FTP (SFTP). Основная цель всех этих действий — предотвратить несанкционированную утечку информации, взлом соединения и прочие атаки, которые случаются даже в домашней LAN. *OpenSSH* стал важной частью почти всех дистрибутивов Linux, так что у вас он гарантированно есть.

OpenSSH продолжает совершенствоваться, и в версии 6.7 предлагается множество улучшений. С точки зрения пользователя, самым потрясающим из них является поддержка пересылки доменного сокета Unix: это выглядело значительным упущением в предыдущих версиях.

Эта функция позволяет передать доменный сокет Unix на локальной машине на удаленный порт TCP или, наоборот, передать удаленный порт TCP на локальный доменный порт Unix; в обоих случаях используется тот же синтаксис, что и в пробросе портов TCP. Например, удаленная база данных *PostgreSQL* может соединяться через безопасный канал SSH

«Появилось возобновление прерванных загрузок в SFTP.»



➤ SDF.org — публичный сервер для работы с каналами SSH.

с доменным сокетом Unix на локальной машине с помощью

```
ssh -L/tmp/test.site:mydatabase.net:5432 second_server
```

Можно также соединить два локальных доменных сокета Unix через SSH.

Среди других новых функций — поддержка шифрования с помощью *LibreSSL* и возобновление прерванных загрузок в SFTP; при этом часть унаследованного кода была удалена, что помогает оптимизации уже весьма многообещающего приложения.

Если вы настолько нетерпеливы, что не хотите ждать обновленных пакетов для своего дистрибутива, или вам нужна индивидуальная сборка, можете скомпилировать *OpenSSH* из исходника.

Инструмент создания снимков ФС

Snapper

Версия: 0.2.4 Сайт: http://snapper.io

Каждый пользователь Linux был бы счастлив иметь надежный инструмент восстановления системы — желательно получше, чем те, что предлагаются в Windows и OS X, но с не меньшей простотой использования.

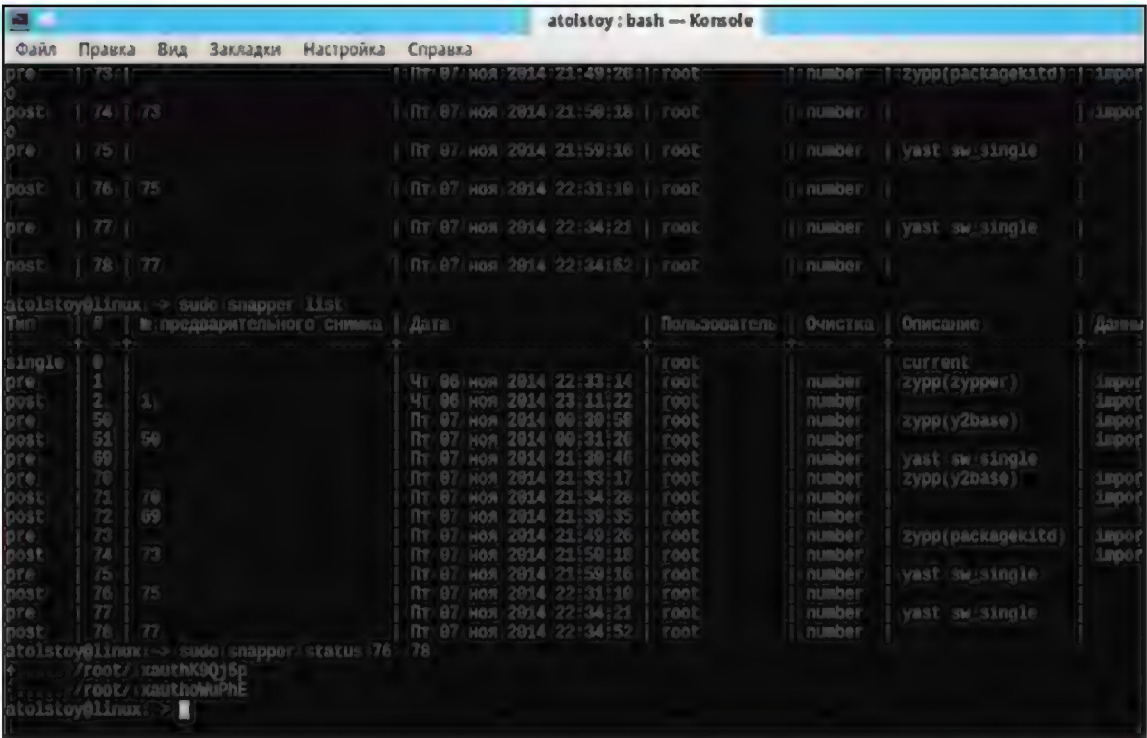
Не исключено, что вы подумали о ZFS от OpenSolaris, однако та почилa в бозе, и хотя по-прежнему активна (и даже портирована в Linux), мало кто применяет ее вне сервера. Btrfs — более современная и, что более важно, стабильная и доступная файловая система для основных версий Linux. Она предлагает много ZFS-подобных функций, делая возможным отслеживание модификаций файловой системы через саму файловую систему, что снимает необходимость в традиционном резервном копировании файлов.

Snapper — это инструмент с открытым кодом, разработанный инженерами SUSE для управления моментальными снимками btrfs; он позволяет просматривать сделанные в системе изменения и отменять

их, возвращаясь к исходному состоянию. Snapper изначально был создан для систем SUSE и openSUSE, однако к ним не привязан; инструмент можно скомпилировать для любой системы Linux. Единственное, к чему он привязан — это к используемой вами файловой системе. И хотя в нем присутствует экспериментальная поддержка ext4, основная цель Snapper в том, чтобы быть интерфейсом к основным функциям btrfs. Чтобы протестировать его, желательно иметь хотя бы один раздел жесткого диска, отформатированный в btrfs, с которым можно поиграть.

Предположим, вы успешно установили Snapper, и вы хотите отследить изменения в своей директории home. Для этого вы используете

«Snapper позволяет просматривать изменения и отменять их.»



➤ Snapper может отменить изменения в вашей файловой системе почти с той же легкостью, с какой используется Ctrl+Z в рабочих приложениях.

```
snapper -c my_home_volume create-config /home'
```

Как видите, Snapper позволяет задать отдельный том для любой директории, чтобы у вас было несколько томов и подтомов на разделе. Для просмотра изменений используйте **snapper list**, и вам будет представлена таблица, где записан каждый моментальный снимок системы. У каждого из них — собственный номер, используемый для работы с ним. Вот как, например, оценить разницу между недавними снимками и вернуться к старому:

```
snapper diff 76..78
snapper undochange 76..78
```

Работает прекрасно. Кстати, многие считают btrfs стабильной, и Snapper — еще одна причина перейти на нее.

Клиент удаленного рабочего стола

Remmina

Версия: 1.1 Сайт: http://remmina.sourceforge.net

Remmina — это инструмент для доступа к удаленным системам через графический интерфейс пользователя. Этот проект конкурирует с проприетарным TeamViewer.

Remmina может напомнить некоторым KRDC, и эти программы в самом деле похожи, но Remmina больше подходит для рабочих столов на основе GTK (Gnome, Cinnamon и Unity), поскольку выглядит аккуратнее и не тянет за собой груз зависимостей KDE.

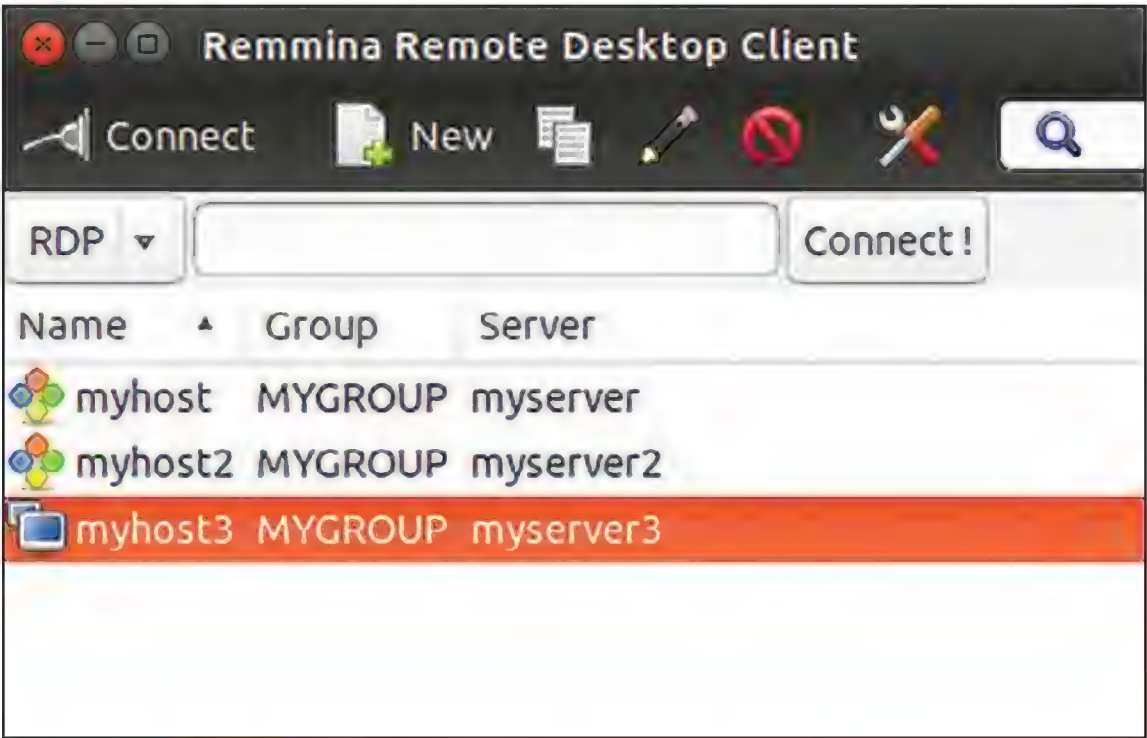
Remmina — разностороннее приложение GTK3, которое включает несколько протоколов соединений, например, RDP, SFTP и SSH, и поддерживает системы Telepathy и VNC. Основное приложение Remmina обычно вносится в пакет отдельно от своих плагинов — тут-то вы и сможете сделать индивидуальную настройку своей системы.

Разработчики Remmina вложили немалые усилия в то, чтобы сделать

приложение дружелюбнее к пользователю. Remmina может поддерживать список профилей соединения, организованных по группам; автоматически прокручивать удаленный рабочий стол большого размера; и также предлагает симпатичную плавающую панель инструментов в окне удаленного рабочего стола, с кнопками переключения и управления.

Изначально идея Remmina заключалась в предоставлении удобной возможности доступа к удаленному компьютеру с Eee PC. Это означает скромное разрешение экрана, поэтому доступ к более крупным рабочим столам приводит к проблемам со скроллингом и перетаскиванием: проблемам, которые Remmina решает. Она

«Включает RDP, SFTP и SSH, поддерживает Telepathy и VNC.»



➤ Главное окно Remmina простое, и разработано, чтобы показывать множество соединений в аккуратной таблице.

не только включает автоскроллинг в режиме окон, но также может похвастаться графическим полноэкранным режимом, где рабочий стол автоматически прокручивается, когда мышь выходит за границы экрана.

Новый релиз 1.1 теперь совместим с FreeRDP 1.1 (одним из движков Remmina) и предлагает ряд других более мелких улучшений, например, по части курсора мыши.

Страница wiki проекта предлагает руководства по компиляции Remmina из исходника в ряде дистрибутивов Linux, но в целом процесс довольно прост.

Файловый менеджер

Wal Commander

Версия: 0.18 Сайт: <http://wcm.linderdaum.com>

Хотя дни *Norton Commander* давно прошли, многие пользователи по-прежнему предпочитают двухпанельные файловые менеджеры. Может быть, вы слышали о *Midnight Commander*? [См. HotPicks **LXF185**, стр. 100.] Это знаменитый Unix-клон прежней оболочки *Norton*, и он настолько классический, что требует запуска из командной строки. *Wal Commander GitHub Edition* немного другой: это подлинное приложение X11. (Довесок “GitHub Edition” означает, что он произошел от первоначального *Wal Commander*.)

Хотя привязка клавиш у *Wal Commander* классическая (F5 для копирования, F8 для удаления, и т.д.), от старомодной двухпанельной оболочки в нем остался только вид. Приступив к изучению функций менеджера, внутри вы обнаружите кучу мощных инструментов. Например, текстовый редактор по умолчанию (доступный по F4) поддерживает выделение синтаксиса для всевозможных языков (C,

C++, sh, Perl, PHP, XML, HTML, Pascal, SQL) и может преобразовывать символы в другую кодировку. Командная строка принимает информацию в синтаксисе DOS и Bash.

Wal Commander также хорош для коммуникаций, при встроенной поддержке SMB и FTP/SFTP, и идеально прост в навигации с помощью стандартных горячих клавиш Alt+<буква> для управления с помощью выделенных букв.

В принципе, по внешнему виду и работе *Wal Commander* похож на приложения Windows типа *Far Manager*, что подтверждается во многих аспектах. Например, можно настроить его на отображение пробела с помощью точек, переключение значков папок, автодополнение, выбрать способ

«Старомодным остался только вид; внутри — мощные инструменты.»



» *Wal Commander GitHub Edition* — не самый элегантный файловый менеджер, но очень удобен и обладает множеством современных функций.

отображения скрытых файлов и, конечно, изменить настройки отображения, в том числе шрифты, цвета и кнопки.

Как и предполагает его название, *Wal Commander GitHub Edition* можно найти в официальной репозитории Git, но есть и множество прекомпилированных двоичных пакетов (и Deb, и RPM), доступных для скачивания, так что вас никто не заставляет компилировать его из исходника, если только вы сами этого не захотите.

Приложение монолитное, и хранит все в своем основном двоичном файле **wcm**. Документация включена в пакет, и если вы впервые используете двухпанельный файловый менеджер, советуем сначала с ней ознакомиться.

Клиент Torrent

rTorrent

Версия: 0.9.4 Сайт: <http://rakshasa.github.io/rtorrent>

На сей раз нас заинтересовал *BitTorrent*, поэтому представляем вам *rTorrent*: torrent-клиент командной строки. Он основан на библиотеке *libTorrent*, разработанной с ним вместе. Также клиент использует *ncurses* для отображения псевдографических кнопок управления в оболочке.

Цель *rTorrent* — предоставить вам полный контроль над вашим равноправным распределенным доступом к файлам как на удаленных системах, так и где бы то ни было, без *X.org* или графического web-интерфейса.

Добавить *rTorrent* исключительно просто: это крошечное приложение, имеющееся практически во всех дистрибутивах Linux. Проблемы и хитрости начинаются тогда, когда вы начинаете им пользоваться.

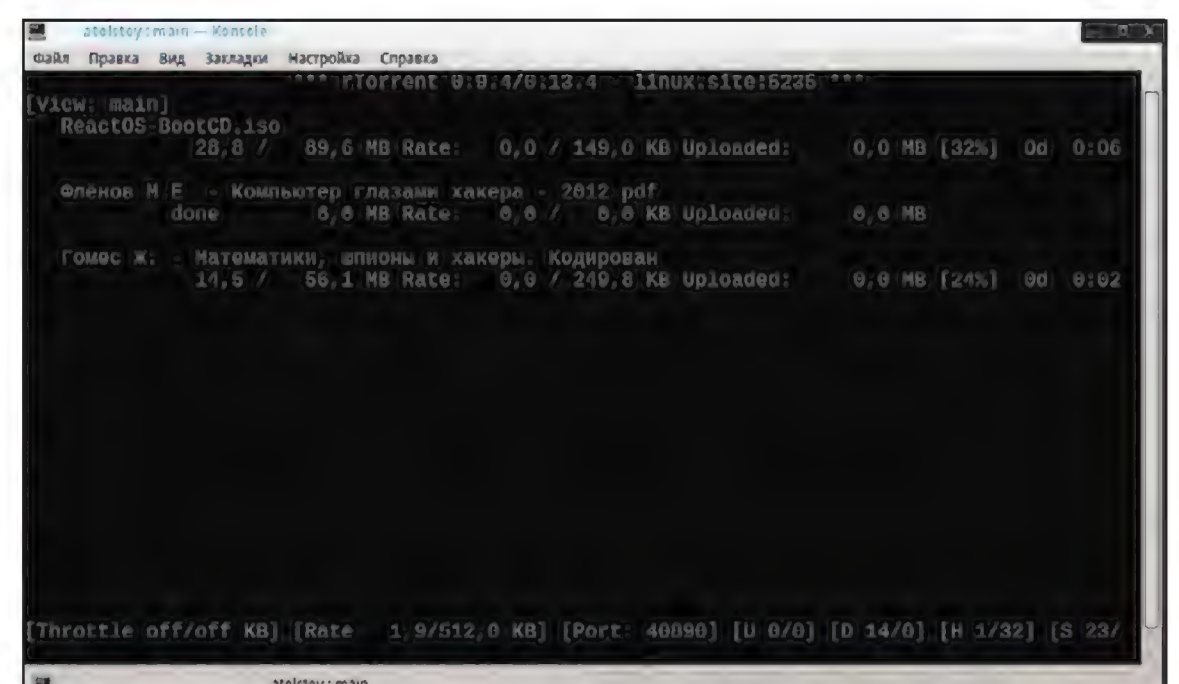
При первом запуске *rTorrent* пожалуется на отсутствие файла `~/rtorrent.rc`. Давайте создадим его вручную. *rTor-*

rent позволяет ознакомиться с внутренней кухней торрентов и указывать десятки основных параметров, так что лучше всего найти учебный (но рабочий) пример на <http://bit.ly/1xgtYig>; длина его 117 строк, и он предоставит только базовые настройки.

Самым очевидным следующим шагом будет изменить настройки по умолчанию, указав папки, где *rTorrent* будет искать новые файлы **.torrent** и где он разместит скачанные. Например, вот так можно настроить *rTorrent*, чтобы он просматривал директорию на предмет поиска новых торрентов и останавливал удаленные:

```
schedule = watch_directory,5,5,load_start=~/rtactive/*.torrent
```

«*rTorrent* позволяет ознакомиться с внутренней кухней торрентов.»



» Скромный черный экран *rTorrent* автоматически заполняется торрентами, обнаруженными в просмотренных директориях.

```
schedule = tied_directory,6,5,start_tied=
schedule = untied_directory,7,5,stop_untied=
```

Другие полезные опции — настройка желаемого минимального и максимального числа пиров; ограничение количества скачиваний и раздач; указание IP-адреса для отображения в трекере.

rTorrent поддерживает множество удобных горячих клавиш — например, маленькие буквы a, s и d для постепенного повышения канала раздачи; а их заглавные эквиваленты увеличивают канал скачивания. Ctrl+S начинает скачивание, а Ctrl+D останавливает его или удаляет остановленное. Чтобы выйти из приложения, просто используйте Ctrl+Q.

HotGames Развлекательные приложения

Платформенная игра

Neverball

Версия: 1.6.0 Сайт: <http://neverball.org>

Если вы играли в классическую настольную игру Labyrinth, вы быстро освоите *Neverball*. В игре надо перемещать большой металлический шар, наклоня игровой мир, и за данный срок провести его по карте, полной препятствий. На вашем пути будут появляться движущиеся платформы и прочие объекты, все сильнее усложняя уровни.

Сначала мы подумали, что игра предназначена для смартфонов с их гироскопами, но оказалось, что *Neverball* впервые увидел свет в 2003 году, а значит, полагается на мышь.

В *Neverball* 116 основных уровней, от простого к сложному, и 25 дополнительных суперсложных уровней. Честно говоря, мы поняли, что можем пройти только «простые» уровни, да и те на настольном ПК, поскольку порт игры для Android на смартфоне оставил нам боль

в руках и смешанные чувства — однако версия для ПК великолепна. По мере перехода на новые уровни ваша связь между рукой и зрением упрочивается, и вы начинаете контролировать малейшие перемещения шара.

Для прохождения уровня надо набрать определенное количество монет и достичь отмеченного желтым выхода. Многие уровни настолько крутые, что нельзя не восхищаться талантом их создателей: сочетание прозрачных кубов, петель, лабиринтов, сложных кривых поверхностей и плавающих платформ, расположенных над бездной и страдающих

«Уровни так круты, что нельзя не восхищаться создателями.»



» Не рассчитывайте, что в *Neverball* все будет просто: каждый уровень заставит вас напрячься, чтобы пройти сложные участки и дыры.

от частых землетрясений и необычной силы тяжести.

Помимо того, что игра затягивает, последняя версия *Neverball* также включает улучшенные пакеты перевода, поддержку «очков дополненной реальности» Oculus Rift и интересные графические улучшения, например, переход к API на OpenGL ES и поддержку дисплеев с высоким разрешением.

Стратегия

O A. D.

Версия: Alpha 17 Сайт: <http://play0ad.com>

Зиро-эй-ди — так нужно произносить ее название — высококачественный клон *Age of Empires II: The Age of Kings*, который появился в 2001 г. В 2003-м разработка переместилась на одиночную игру с собственным движком (*Pyrogenesis*), и в 2010-м игра вышла как свободная программа с открытым кодом.

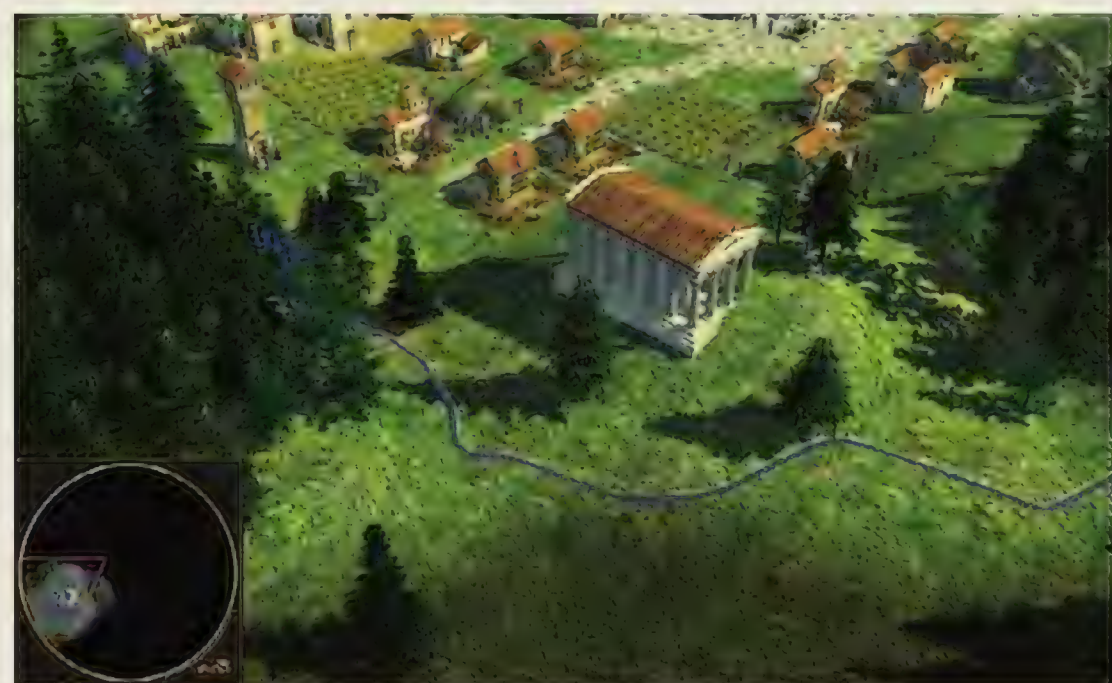
O A. D. воссоздает основные цивилизации с 500 г. до н. э. до 500 г. н. э., а название относится к пропущенному в середине нулю. Хотя официально *O A. D.* все еще на стадии альфа, это вполне играбельная стратегия древних войн в реальном времени (РМВ). Она подготовлена для десятков дистрибутивов Linux и обычно представлена в двух пакетах: игровом движке и солидном архиве данных размером 1,5 ГБ. Из чистого любопытства мы заглянули в пакет *Oad-data* и обнаружили, что как минимум его

половина занята высококачественными текстурами и сетками.

Несмотря на внушительные размеры, игра запускается мгновенно и демонстрирует предупреждение о том, чего не хватает в альфа-версии. Не волнуйтесь: все основные функции отлично работают. Графика очень ровная, и игра очень удобна в управлении благодаря мягкому скроллингу и увеличению масштаба.

Цель игры — постройка древнего города посредством сбора ресурсов (пищи, дерева, металла, камня и т. д.) и постройки зданий и юнитов, как военного, так и гражданского назначения. По мере

«Вполне играбельная стратегия древних войн в РМВ.»



» *O A. D.* — место, где можно совершенно спокойно заниматься хищнической вырубкой отлично смоделированных деревьев.

возможности следует тренировать армию, что может даже включать применение осадных орудий и боевых кораблей. Обучение военных единиц стоит ресурсов, поэтому игра подталкивает вас к единственно возможной жизнеспособной стратегии: сначала стройте инфраструктуру, а уж потом — армию.

Время, отведенное вам на первую стадию, зависит от того, против кого вы играете. Вы можете выбрать одного из разных ИИ-противников (много не выбирайте: это затормозит игру). Если вы играете против одного врага (или «фракции»), у вас будет достаточно времени на постройку города до того, как начнутся военные действия.

Клиент IRC

Konversation

Версия: 1.5.1 Сайт: <https://konversation.kde.org>

Internet Relay Chat (IRC) продолжает оставаться популярной формой текстового общения, особенно среди разработчиков программ. *Konversation*, что и заметно, является частью программного пакета KDE и служит «официальным» интерфейсом IRC для команды KDE. Так что он, возможно, уже установлен в вашей системе; а если нет, то установится парой щелчков по вашему менеджеру пакетов.

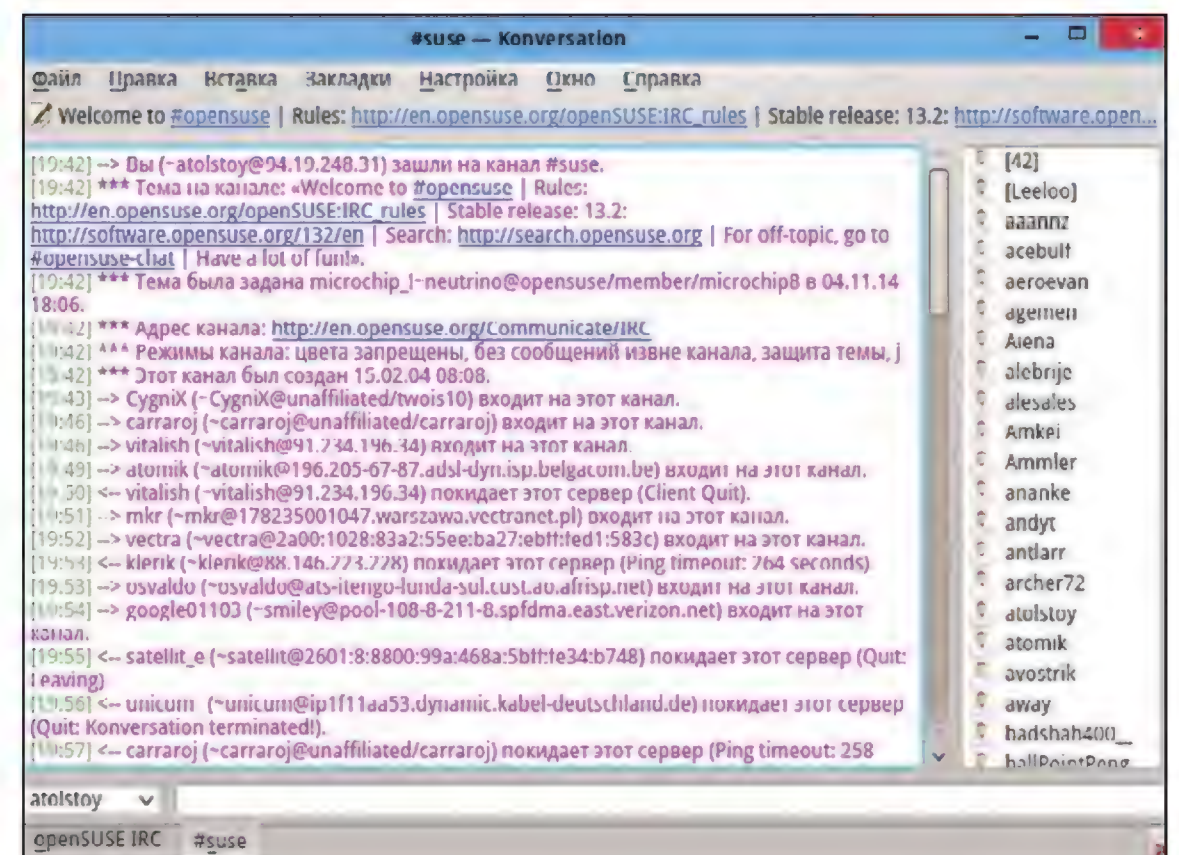
Чтобы использовать IRC, надо соединиться с сервером, а потом присоединиться к каналу (эквивалент чат-рума) или просто к частному разговору. *Konversation* уже при запуске содержит один сервер по умолчанию (нажмите F2, чтобы вывести список настроек сервера) и канал. Стандартный пакет *Konversation* обычно устанавливает в качестве преднастроенного сервера и канала Freenode и #kde, но некоторые разновидности Linux могут индивидуально изменять выбор по умолчанию (как, например, openSUSE, показанный на рисунке справа). Если вы выбрали

соединение с сервером, проверьте правильность вашей информации. *Konversation* автоматически заполняет ваш профиль IRC данными, полученными при входе в вашу систему. Если в прошлом вы не до конца заполнили его (например, пропустили поле полного имени), *Konversation* предложит вам вручную отредактировать свою информацию.

Когда вы, наконец, присоединитесь к разговору, вы увидите основное окно чата и панель со списком пользователей в текущем соединении справа. Вкладки под основной областью перечисляют имеющиеся каналы; вы можете поочередно соединяться с любым числом каналов.

Есть несколько полезных команд IRC, используемых прямо в строке ввода.

«**Konversation автоматически заполняет ваш профиль IRC.**»



► **IRC: удобный способ общения с пользователями Linux.**

Например, введите `/join#имя_канала`, и вы присоединитесь к известному вам каналу. Если такого канала не существует, он создастся, и вы станете его оператором. Если вы — новичок в IRC, можете нажать на F1, чтобы познакомиться с очень удобным справочником *Konversation* для неопытных пользователей.

Konversation отличается исключительной индивидуальной настраиваемостью: можно настроить его функцию выделения для лучшей читаемости чата, звуковые уведомления и многое чего еще. Программа также способна передавать файлы и может синхронизироваться с *KAddressBook*, если вы его используете.

Медиа-плеер

Tomahawk

Версия: 0.8 Сайт: www.tomahawk-player.org

В Linux так много медиа-плееров, что впору ломать голову: каких же функций ещё недостаёт? Разработчики *Tomahawk* полагают, что им это известно, и создают социальный медиа-плеер.

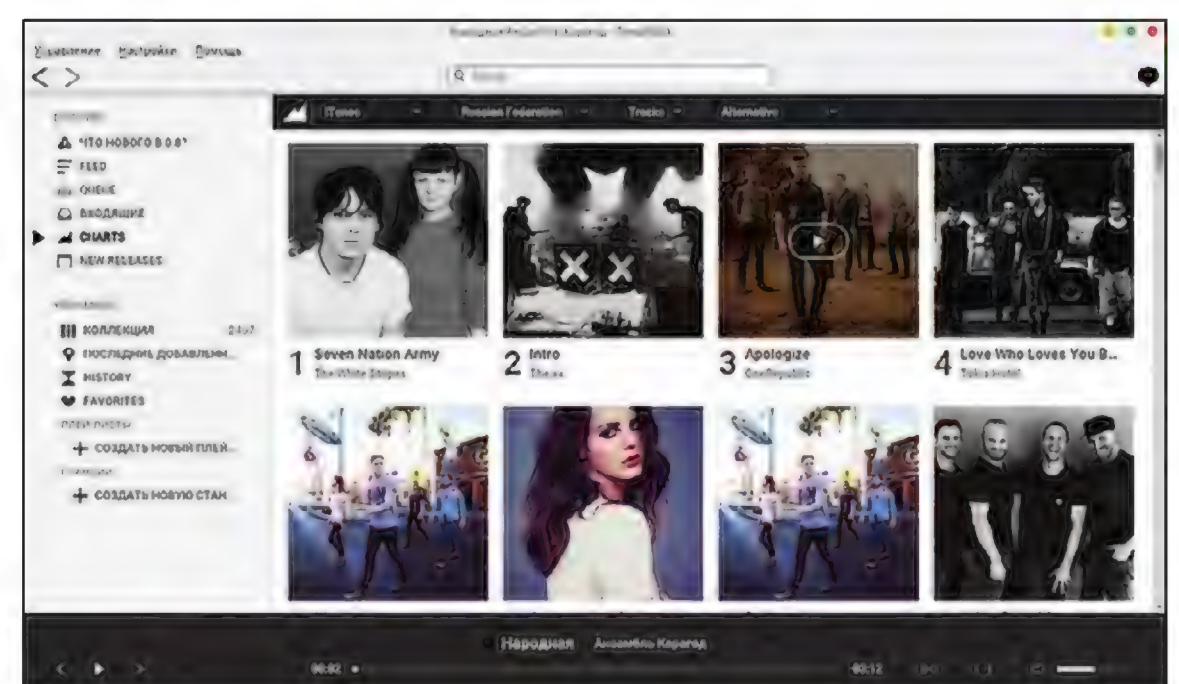
Tomahawk — современное приложение Qt 5 с рядом зависимостей от KDE Frameworks 5 и VLC-noX. Суть плеера в том, чтобы плавно интегрировать разные источники музыки в один солидный менеджер медиа. По этой причине *Tomahawk* предлагает десятки плагинов, расширяющих охват импортируемой музыки. Как и следовало ожидать, можно соединяться со Spotify, SoundCloud, Last.fm, Grooveshark, Jamendo, Subsonic, Beats Music и другими, менее известными сервисами, а также с облачным хранением, например, ownCloud и Google Play Music; и воспроизводить музыку от других пользователей *Tomahawk*. То есть воспроизводить в виде потока контент с распределенным

доступом от локальных участников вашей LAN или удаленно через XMPP (Jabber). Эти плагины предлагают уникальную возможность: здесь практически не отличить локальный контент от потокового.

При запуске *Tomahawk* вас приветствует окно настройки, где вы можете включить нужные плагины из первой вкладки, выбрать локальную музыкальную папку из второй и изменить перенаправление портов и настройки прокси в третьей.

Интерфейс *Tomahawk* представляет собой нечто среднее между почтовым клиентом и настроенной версией iTunes. Раздел Discover на боковой панели предлагает социальные опции, показывая вам вашу ленту и треки, которыми поделились

«**Плавно интегрирует источники музыки в одну коллекцию.**»



► **Tomahawk позволяет просматривать невероятное количество источников музыки, чтобы найти свои любимые мелодии или открыть для себя новые.**

с вами друзья, и позволяя вам взаимно делиться треками с ними (предполагается, что вы поместите свои предложения в очередь).

Локально хранимая музыка живет именно здесь, в разделе Your Music, где вы можете просматривать свою коллекцию и добавлять треки в плей-листы.

Самая верхняя панель плеера применяется для навигации и поиска треков, а нижняя область отображает классическую панель инструментов управления с общими кнопками управления и регулировки громкости — там показано, что в данный момент воспроизводится. **LXF**

На диске

Дистрибутивы, приложения, игры, подкасты и всякое-разное...

Лучшее из Интернета, упакованное в 9 ГБ качественного DVD.



Сдвиг по битам

Дистрибутивы на DVD этого номера — все, кроме одного — 64-битные. Тенденция, однако...

Это вовсе не часть вселенского заговора с целью отвлечь пользователей от (очень) старых машин; это просто прогресс. Ну и так уж вышло в этом месяце. При необходимости мы будем включать в DVD и 32-битные дистрибутивы, но полагаем, что их число уменьшится.

Мы уже видим дистрибутивы, выходящие только в 64-битных версиях, например, XBMCbuntu. В этом есть смысл: вряд ли ПК десятилетней давности — хороший выбор для воспроизведения HD-мультимедиа. Все рабочие столы обрастают функциями и извлекают преимущества из большего объема ОЗУ и ускорения процессоров, и 32-битные версии стали менее востребованы.

Мы не говорим, что для 32-битных дистрибутивов нет места в экосистеме Linux или на Linux Format DVD. Некоторые дистрибутивы идеальны, и даже специально предназначены для более старого оборудования. Они и дальше будут выпускать 32-битные версии, а мы продолжим о них рассказы. Однако на дворе как-никак 2015 год, и пора прощаться с частью старых систем, даже если мы их нежно любим...

Neil

» Важно ВНИМАНИЕ!

Порченые диски

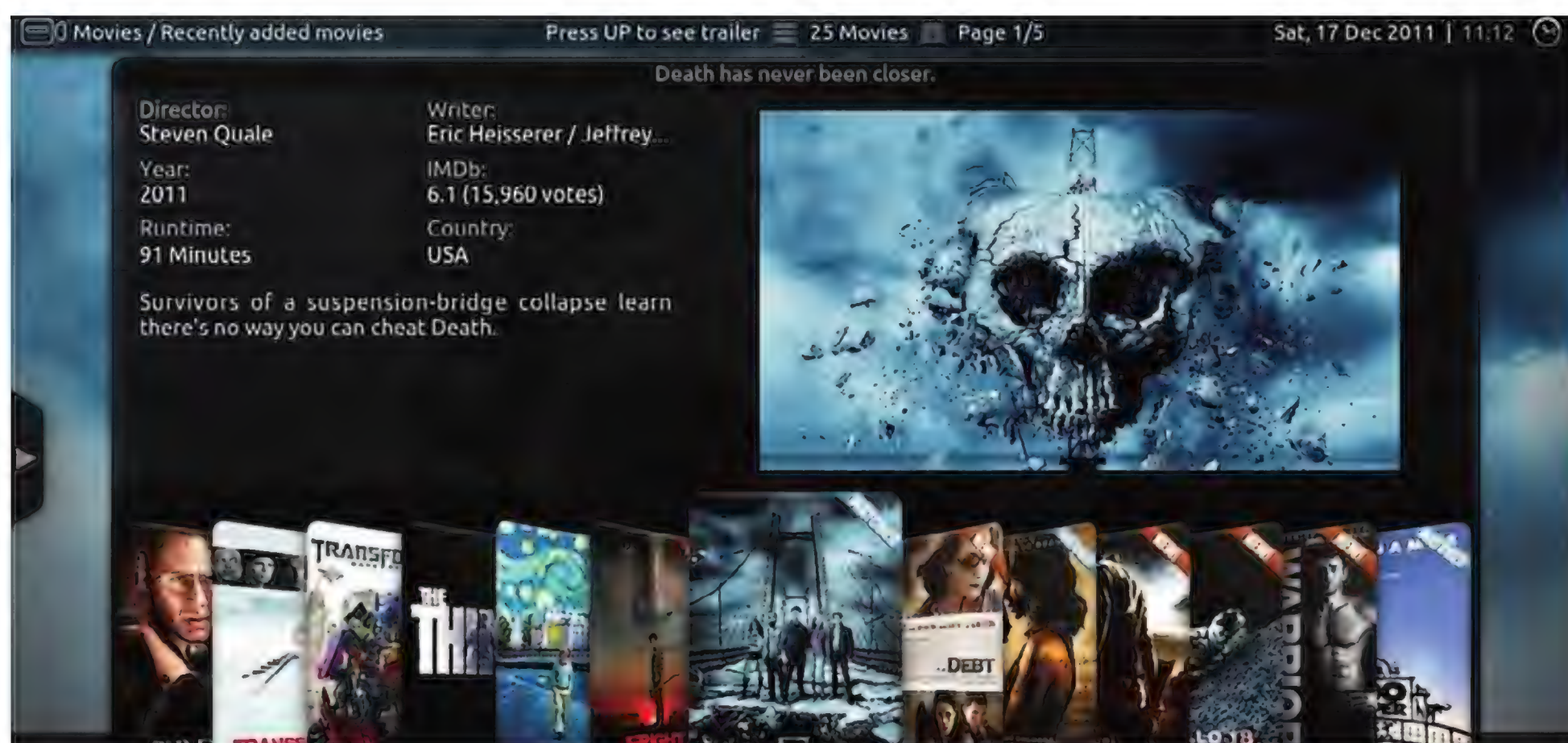
В маловероятном случае какого-то дефекта вашего **LXF DVD** обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

Друг киноманов

XBMCbuntu 13.2

В этом месяце у нас статья про XBMC, и мы решили, что должны дать вам шанс с ним познакомиться. По сути, XBMCbuntu — это Lubuntu с предустановленным XBMC. Когда вы загружаете среду live, она переходит прямо в XBMC. Желая его установить, выйдите из XBMC и выберите XBMCbuntu

в меню вверху справа на экране, а потом снова войдите в систему как пользователь xbmc без пароля. Появится рабочий стол LXDE, откуда вы сможете запустить программу установки. XBMC откажется загружаться, если ваша видеокарта или драйвер не подходят. Тогда вы увидите только экран приглашения.



Из первых сторонников btrfs

openSUSE 13.2

Мы считаем Ubuntu дистрибутивом-ветераном, однако openSUSE почти в два раза старше. Изначально основанный на Slackware, он вскоре перешел на RPM и стал независимым

дистрибутивом. Подобно Red Hat, SUSE имеет две основные линии. Дистрибутивы для предприятия — SLE (SUSE Linux Enterprise), а проект сообщества — openSUSE. SUSE долго поддерживал рабочий стол KDE, и именно такая версия — на нашем DVD; однако предусмотрены и другие рабочие столы, их можно установить после установки дистрибутива.

Одна из самых примечательных функций SUSE — YaST, что означает Yet another Setup Tool [Очередной инструмент настройки], но на самом деле он представляет собой нечто большее. Это интегрированный интерфейс администрирования, способный справиться со всем, от первичной установки до управления пакетами, от настройки сети до управления оборудованием. И в openSUSE есть не только YaST: это завершённый и зрелый дистрибутив, который не боится пробовать новые технологии, и первый дистрибутив, принявший btrfs как файловую систему по умолчанию.





Новичок в Linux? Начните отсюда!

- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент *MS Office*?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы приводятся в [Index.html](#) на диске.

Свежая ветка Canonical

Ubuntu 14.10

Ubuntu 14.10 появляется на нашем диске не впервые, но он соответствует содержанию данного номера журнала. Раньше мы делали наши ремиксы с несколькими дополнительными рабочими столами, а этот — с базовым (правда, базовый не то чтобы всегда был пурпурного цвета) рабочим столом Ubuntu Unity. Установка у него компактнее и быстрее, и после нее вы сможете добавить любые рабочие столы по своему усмотрению, запустив поиск в менеджере пакетов на предмет наличия соответствующих пакетов, например, *xubuntu-desktop* для Xfce или *kubuntu-desktop* для KDE.



Элегантный медиа-центр

OpenELEC 4.95.3

OpenELEC — еще один дистрибутив на базе XBMC, на сей раз созданный с нуля, поэтому он намного меньше и занимает меньше места на диске — хотя вряд ли это будет проблемой, раз вам понадобятся крутые гигабайты для вашего медиа-центра: что для него несколько сотен

мегабайт? Дистрибутив не предлагается в виде ISO-образа, и на DVD вы найдете файл TAR. Распакуйте его на свой жесткий диск, перейдите по **cd** в содержащую его папку, воткните USB-флэшку и запустите `sudo ./create_installstick /dev/sdX`

где sdX — ваш диск USB. Если вы не уверены в том, какое это устройство, то, вставив флэшку, скомандуйте `dmesg | grep 'removable disk'`

После отработки `create_installstick` вы сможете загрузиться прямо с этой флэшки для установки OpenELEC.

Если в вашей целевой системе нет DVD-привода, для создания флэшки можно запустить пакет на другом ПК. На DVD предусмотрен также BAT-файл Windows, чтобы вы могли проделать то же самое даже на машине с Windows. **LXF**



И еще!

Системные инструменты

Главное

Checkinstall Установка tar-архива с помощью менеджера пакетов.

GNU Core Utils Основные утилиты, обязанные присутствовать в каждой операционной системе.

Hardinfo Инструмент для тестирования системы.

Ядро Исходник последней версии ядра.

Memtest86+ Проверьте свое ОЗУ на сбои и ошибки.

Plop Простой менеджер загрузки для запуска операционных систем.

RaWrite Создавайте загрузочные диски в Windows.

SBM Независимый от ОС менеджер загрузки с простым в использовании интерфейсом.

WvDial Соединяйтесь с Интернетом через телефонный модем.

Чтение

Книжная полка

The Cathedral and the Bazaar [Собор и Базар] Классический текст Эрика С. Реймонда [Eric S Raymond], объясняющий преимущества открытой разработки.

Linux Kernel in a Nutshell [Ядро Linux в двух словах] Введение в ядро, написанное магистром хакеров Грегом Кроа-Хартманом [Greg Kroah-Hartman].

Справочник администратора Debian Руководство для системных администраторов.

Словарь Linux Linux от A до Z.

Окунитесь в Python Мастер-класс по этому популярному языку.

Руководство по созданию скриптов Bourne Shell Начинайте осваивать скрипты оболочки.

Расширенное руководство по созданию скриптов Bash Больше информации для опытных пользователей.

Intro to Linux Руководство для новичков.

Руководство сисадмина Основы работы с системой.

Обзор утилит Обзор инструментария GNU.

Пропустили номер?



Закажите его через сайт www.linuxformat.ru в «ГНУ/Линуксцентре»! Журналы доставляются и в печатной, и в электронной форме, так что с момента открытия браузера до получения нужного вам выпуска LXF может пройти всего пара минут!

Прямо сейчас для заказа доступны следующие номера:

LXF189
Ноябрь 2014



250 руб.

- » Нет проблем! Или — решим легко
- » Ваш портативный друг
Выбираем Chromebook
- » Прошлое криптографии
Как человечество хранило секреты
- » Осваиваем макросы Автоматизация офиса

LXFDVD: Tails 1.1 Live OS, Deepin 2014, Rescatux 0.32b1, System Rescue CD 4.3.0, Ultimate Boot CD 5.3.0, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_189/
PDF-версия: shop.linuxformat.ru/elxf_189/

LXF190
Декабрь 2014



250 руб.

- » Кто на свете всех милее?
Дистрибутив-2014
- » Жизнь в Сети Браузер вашей мечты
- » PhotoRec Чьих рук дело — спасение файлов
- » Pi и Mathematica Две знаменитости

LXFDVD: Linux Mint Debian Edition, Trisquel, Mageia, CentOS, Korora, openSUSE, Salix, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_190/
PDF-версия: shop.linuxformat.ru/elxf_190/

LXF191/192
Январь 2015



250 руб.

- » Пиршество идей Проекты для Raspberry Pi
- » Звуки музыки Плееры для фонов
- » Робот — друг человека
Соберем для дома, для семьи
- » Забыли пароль? Rescatux вас спасет

LXFDVD: все для Raspberry Pi — Raspbian, KaliLinux, PiMusicBox, RetroPie, Jasper, NOOBS, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_191-192/
PDF-версия: shop.linuxformat.ru/elxf_191-192/

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через www.linuxformat.ru/subscribe или www.linuxcenter.ru, получают электронную версию в подарок! На сайте shop.linuxformat.ru вы также сможете приобрести предыдущие выпуски LXF.

Подписывайтесь на сайте www.linuxformat.ru/subscribe

Телефоны отдела подписки:

- » Санкт-Петербург (812) 309-0686
- » Москва (499) 271-4954



Linux Format ВКонтакте

Вступайте в нашу
группу vk.com/linuxform

На странице LXF ВКонтакте вы найдете:

- » Новости о Linux
- » Статьи из архива LXF
- » Анонс свежего выпуска LXF и часть статей из него
- » Живое общение и онлайн-консультацию по подписке на наше издание

Информация о диске

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: “tgz” — это сокращение от “tar.gz”;
- » **имя_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Внимательно прочтите это перед тем, как использовать LXFDVD!



» Содержание

ДИСТРИБУТИВЫ

OpenELEC 4.95.3 (64-битный, ISO-образ)
openSUSE 13.2 (64-битный, ISO-образ, загрузка с **LXFDVD**)
Ubuntu 14.0 (64-битный, ISO-образ, загрузка с **LXFDVD**)
XBMCbuntu 13.2 (64-битный)
NandyLinux (32-битный, ISO-образ, загрузка с **LXFDVD**)
Netrunner (64-битный, загрузка с **LXFDVD**)
Robolinux (64-битный, ISO-образ, загрузка с **LXFDVD**)

НОТРИКСЫ

0 A.D.
CherryTree
Konversation
Lunar Linux
Neverball
OpenSSH
Remmina
rTorrent
Snapper
Tomahawk
Wal Commander GitHub Edition

СРАВНЕНИЕ: ВЕКТОРНЫЕ РЕДАКТОРЫ

Inkscape
Karbon
LibreOffice Draw
sk1
Xara Xtreme

УЧЕБНИКИ

Minix 3
Файлы для урока по Python
Файлы для урока по Minetest
ПО для очистки диска
Файлы для урока по ПЛИС

ПОМОЩЬ

Руководство новичка
Руководства
Ответы

Пожалуйста, перед использованием данного диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru
ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь, пожалуйста, по адресу disks@linuxformat.ru

LINUX FORMAT

Dive Into Python Учебник по программированию на языке Python

Intro to Linux Начальное руководство по Linux

Linux Dictionary Словарь Linux, объясняющий специфическую терминологию

Linux Kernel in a Nutshell Описание ядра Linux, созданное одним из его выдающихся разработчиков — Греггом Кроа-Хартманом [Greg Kroah-Hartman]

System Administrators Guide Руководство по базовому администрированию Linux

GNU Tools Summary Руководство по работе в командной строке и обзор основных утилит GNU

ДОКУМЕНТАЦИЯ — 10 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)

Bash Scripting Подробное руководство по программированию на Bash

Bourne Shell Scripting Начальное руководство по программированию на Bash

Cathedral Bazaar Классический текст Эрика Раймонда [Eric S Raymond] «Собор и базар»

The Debian Administrator's Handbook Руководство администратора, написанное разработчиками Debian

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства; однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.
Тираж изготовлен ООО «Марком», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юкки, Школьная ул., 7-а. Лицензия МПТР ВАФ N 77-103.

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права супер-пользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле */etc/default/cdrecord*. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.



UNIXFORUM.org



Есть вопросы?
Задавайте!

<http://unixforum.org/>

Отдел дистрибьюции ГНУ/Линуксцентра приглашает дилеров и дистрибьюторов к сотрудничеству!

Широкая сеть представительств
в разных городах России
позволит вам оптимизировать
процессы логистики и доставки товара

ПОДРОБНЕЕ О ПАРТНЕРСКОЙ ПРОГРАММЕ:
WWW.LINUXCENTER.RU/PARTNER/



ДИСТРИБУТИВЫ GNU/LINUX НА USB FLASH



Linux Mint 16



Ubuntu 14.04

А ТАКЖЕ
версии для юриди-
ческих лиц —
с лицензионным
договором
присоединения

USB Flash 8 Gb

495 рублей

www.linuxcenter.ru/shop/distros/usb/

RH124, RH134, RH254
февраль 2015



RHEL7 в формате LVC

Пройди путь от новичка до инженера Red Hat не выходя из дома

**LVC (LIVE VIRTUAL CLASS) -
ИНТЕРАКТИВНОЕ ДИСТАНЦИОННОЕ ОБУЧЕНИЕ
В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ**

RH124 - RED HAT SYSTEM ADMINISTRATION I	02.02.15 - 06.02.15
RH134 - RED HAT SYSTEM ADMINISTRATION II	09.02.15 - 12.02.15
RH254 - RED HAT SYSTEM ADMINISTRATION III	16.02.15 - 19.02.15

WWW.UNIXEDU.RU

ГНУ/Линуксцентр
приглашает на работу!



ВАКАНСИЯ: Разработчик систем электронного документооборота

ТРЕБОВАНИЯ:

- » Опыт внедрения и сопровождения систем электронного документооборота на базе Alfresco
- » Умение описать бизнес-процесс
- » Знание SQL, понимание принципов построения и функционирования баз данных
- » Знание СЭД, отличных от Alfresco, приветствуется

ОБЯЗАННОСТИ:

- » Разработка систем с нуля, сопровождение
- » Настройка системы в соответствии с требованиями бизнеса
- » Написание инструкций для конечных пользователей

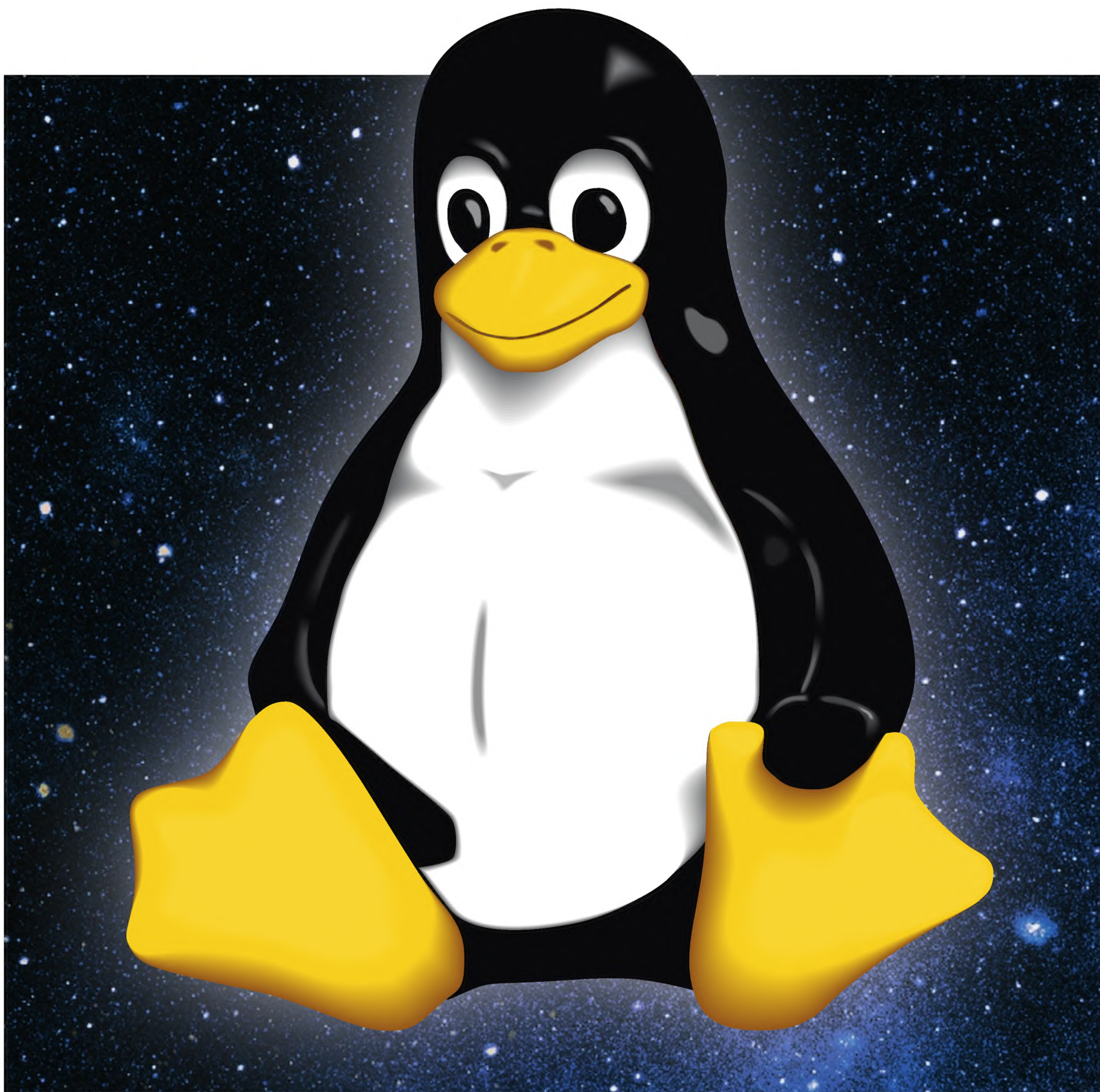
ПОДРОБНЕЕ: www.linuxcenter.ru/vacancy/

Футболки GNU/Linux

БОЛЕЕ 100 ВИДОВ. ВСЕ РАЗМЕРЫ И ЦВЕТА



WWW.LINUXCENTER.RU/SHOP/GIFTS/



В мартовском номере Строим Linux- систему нового поколения

Шагаем в авангарде! Исчерпывающее руководство по суперсовременной графике, файловым системам и прочему.

Мышь на Micro Python

Полный проект: превратим плату в USB-мышь, пользуясь версией Python 3 с микропроцессорным уклоном.

Лучший рабочий стол Linux

Тест-групповуха — от Cinnamon до Enlightenment: выбираем лучшую рабочую среду, предлагаемую Linux.

Стать гитаристом-иконой

Озвучьте своего внутреннего Сантану через виртуальный усилитель *guitarix*, добавив эффектов в гитарное соло.

Содержание будущих выпусков может меняться — вдруг нас задержат сторожа хранилища...

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж печатной версии 3000 экз., распространение электронной версии 30000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор

Кирилл Степанов info@linuxformat.ru

Литературный и выпускающий редактор

Елена Толстякова

Переводчики

Максим Алибаев, Елена Ессяк, Даниил Кривошеин, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова

Редактор диска

Александр Баракин

Верстка, допечатная подготовка

Сергей Рогожников

Технический директор

Андрей Смирнов

Директор по рекламе

Владимир Савельев advert@linuxformat.ru

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатель

ООО «Линукс Формат»

Отпечатано в типографии ООО «ЛД-ПРИНТ»

196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. б/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru
Заказ 12171

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com

Научный редактор Джонни Бидвелл [Jonni Bidwell]

jonni.bidwell@futurenet.com

Выпускающий редактор Крис Торнетт [Chris Thornett]

chris.thornett@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоза

[Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ

Рассел Барнс [Russell Barnes], Даниил Батурич [Daniil Baturin], Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Шон Конвей [Sean Conway], Том Хэтфилд [Tom Hatfield], Кевин Ли [Kevin Lee], Эндрю Моллетт [Andrew Mallett], Ник Пирс [Nick Peers], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Ричард Смедли [Richard Smedley], Мэтт Свайдер [Matt Swider], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Евгений Балдин, Андрей Гондаренков, Михаил Остапкевич, Алексей Федорчук

Поддержка выпуска Алекс Саммерсби [Alex Summersby],

Джим Тэkker [Jim Thacker]

Художественный ассистент Элвин Уитмен [Alvin Wheatman]

Иллюстрации Шейн Коллиндж [Shane Collinge],

Саймон Миддлвик [Simon Middleweek]

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 01225 442244, email: linuxformat@futurenet.com

РОССИЯ:

Санкт-Петербург (редакция):

пр. Медиков, 5, корп. 7. Тел. +7 (812) 309-0686

По вопросам сотрудничества, партнерства, оптовых закупок:

partner@linuxcenter.ru

Авторские права: статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственности за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds]. "GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт www.futureplc.com.



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

Новое поколение средств защиты

Межсетевые экраны ССПТ, не имеющие IP-адреса

ССПТ-2 — это сертифицированное ФСТЭК, ФСБ и ГАЗПРОМСЕРТ средство защиты информации нового поколения, реализующее функции межсетевого экрана, но при этом остающееся «невидимым» для любых протоколов и тестовых воздействий, что достигается за счет отсутствия физических и логических адресов на его фильтрующих интерфейсах. ССПТ-2 невозможно обнаружить никакими известными средствами удаленного мониторинга сети.

Скрытность функционирования межсетевого экрана повышает надежность системы защиты в целом и существенно упрощает процедуру установки ССПТ-2 в компьютерные сети и функционирующие на их основе информационные и телематические системы.

Защита для высокоскоростных корпоративных сетей Ethernet 100/1000 Мбит/с

Сертифицированы ФСТЭК и ФСБ (3-й класс защиты)

На базе процессоров с 64-разрядной многоядерной архитектурой



Назначение устройства

Основное средство защиты для реализации различных политик информационной безопасности с помощью:

- фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;
- управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);
- контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

- обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;
- мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;
- обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux  center

HETZNER DEDICATED SERVERS

HETZNER — ONLINE —

В ЦЕЛОСТИ И СОХРАННОСТИ!

ХРАНЕНИЕ ДАННЫХ В ГЕРМАНИИ



ВЫДЕЛЕННЫЙ СЕРВЕР PX60

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 2 ТБ 6 Гбит/с SATA 7200 об/м Enterprise класс Software-RAID 1
- 30 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

2600

рублей в месяц
+ Установка 3700 рублей



ВЫДЕЛЕННЫЙ СЕРВЕР PX70-SSD

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 480 ГБ 6 Гбит/с SATA SSD Data Center Series
- 50 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

3700

рублей в месяц
+ Установка 3700 рублей



Безопасное защищённое хранение ваших данных в собственных дата-центрах немецкой компании Hetzner Online, соответствующее высоким требованиям к стандартам защиты данных в Германии.

RU.HETZNER.COM



100%
Green Electricity
Energy-efficient
Hardware

GreenIT **2011**
Best Practice Award

Hetzner Online активно поддерживает защиту окружающей среды, используя исключительно 100% возобновляемые источники энергии. Выбирайте более чистое будущее вместе с Hetzner Online!

* Нет платы за превышение. При превышении 30 ТБ/месяц (PX60), 50 ТБ/месяц (PX70-SSD) скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 80 рублей за каждый дополнительный ТБ.